BL0938 产品技术规范

1. 目的和使用范围

本规范介绍了 BL0938 产品的特点，基本功能，电路原理，并介绍了电路的典型应用情况。

2. 产品简述

2.1 基本特性

BL0938 是上海贝岭股份有限公司开发的高速低功耗 8 位微控制器单元（MCU）系列中的一个产品。它含片内存储单元包括 3840 个字节用户 ROM 和 192 个字节用户 RAM,采用 HCMOS 工艺技术制造。BL0938 的指令系统与美国 MOTOROLA 公司的 MC68HC05 系列完全兼容。为适应不同的用户需求，在原有 BL2220 的基础上进行了一些功能增减；清除了 PA 口、PD 口（A/D 转换器）; 增加了 38K 红外调制输出功能、UART 串行通讯接口、内部可编程的实时时钟 RTC。

BL0938 主要适用于多功能电子电度表（包括复费率、预付费及集中抄表系统）和其它嵌入式民用及工业控制系统。

2.2 电路框图及特点

特点:

- 指令系统与 BL2220 完全兼容。
- 低功耗；异步全静态的芯片设计。

  ——支持 STOP，WAIT 省电方式。
- 用户 ROM 4K 字节(含 240 字节自检 ROM)。
- 192 字节 RAM(含 64 字节堆栈)。
- 八位定时器，带七位预定标器。

- 双路片内晶振电路。

  ——4MHz，作为内部系统总线的时钟。

  ——32.768KHz（或 32KHz），作为实时时钟。

- 两个通道各 8 个双向 I/O 口(接通电源时都为高阻，即输入状态)。

  ——每个 IO 端口都有可编程上拉电阻。

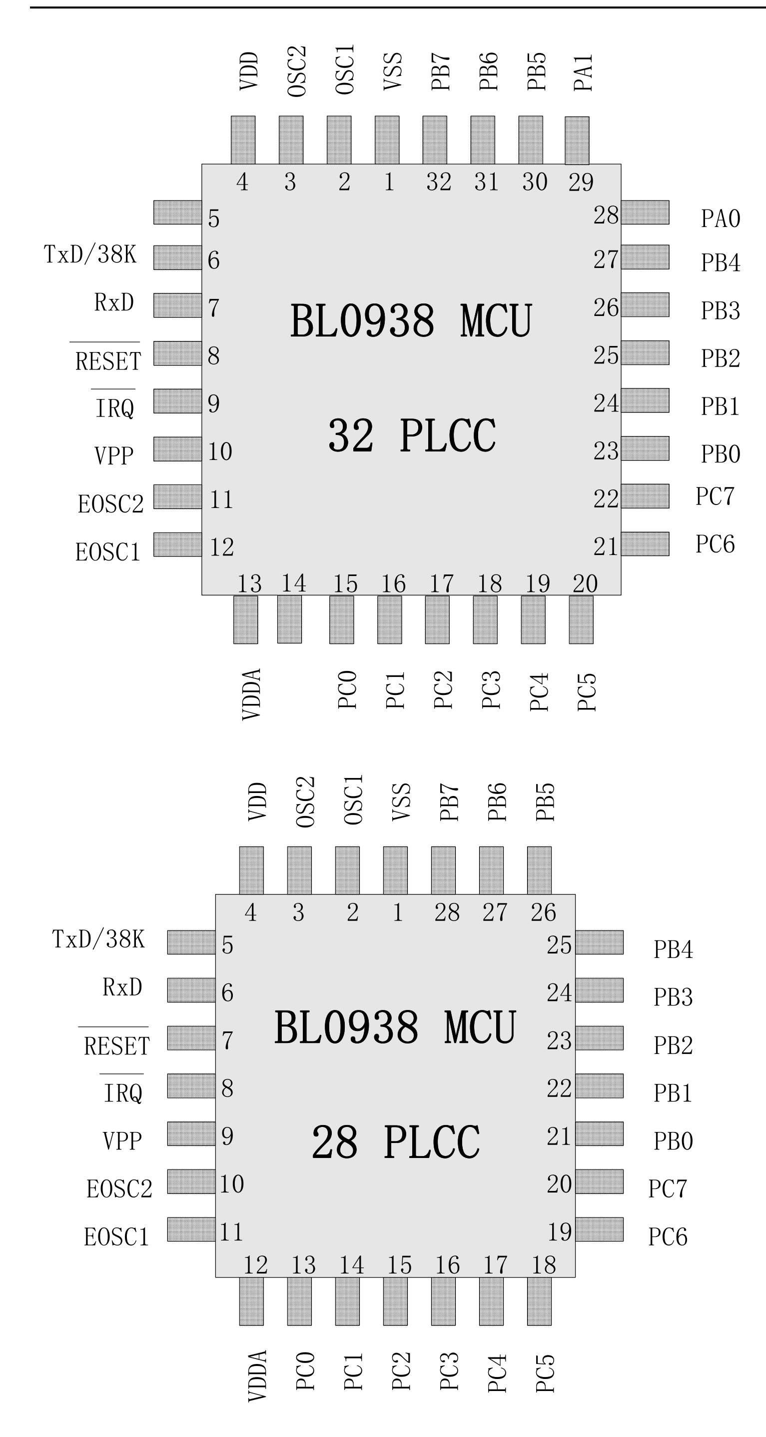
  ——每个 IO 端口能驱动标准 TTL 电路的电流。

- 内含硬件看门狗电路(Watchdog)。
- 含有七个中断源：定时器中断、软件中断、外中断、RTC 中断、UART 中断、低电压检测中断、键盘中断。
- 有一路 UART 串行通讯接口。
- 具有可编程的实时时钟 RTC，可以产生一秒或十秒中断。
- IRQ，RESET 内部有 100K 上拉电阻。
- 标准 5 伏电压工作。系统掉电时，自动切换至 3.3 伏备用电池，对 RTC 供电。此时功耗小于 1μA。
- 多种复位功能包括：低电压复位，上电复位，Reset 管脚复位，Watchdog 复位。
- PLCC28、PLCC32 封装方式。

BL0938 MCU 的结构框图如下:

USER MASK ROM – 3840 BYTES
SELF-CHECK ROM – 240 BYTES
SRAM – 192 BYTES
KEYBOARD INTERRUPT
DDR A
PORT A
8
PA0~PA1
ACCUMULATOR
INDEX REGISTER
STACK POINTER
PROGRAM COUNTER
CONDITION CODE REGISTER
DDR B
PORT B
PB0~PB7
DDR C
PORT C
PC0~PC7
IRQ
RESET
RESET
BUS-CLOCK
VOLTAGE RESET
POWER
VDD
VSS
OSC1
OSC2
OSC/2
7-BIT PRESCALER
8-BIT COUNTER
TIMER CONTROL
COP WATCHDOG
UART
TXD
RXD
Real Timer Clock
EOSC2
EOSC1
VDDA

2.3 管 脚 图:



2.4 管脚功能定义及说明

PLCC28 封装方式:

PIN NO	PIN NAME	FUNCTION	VALUE	IN/OUT
1	VSS	电源地	0V	In

PIN NO	PIN NAME	FUNCTION	VALUE	IN/OUT
2	XTAL1	振荡器引脚	4MHz	In
3	XTAL2	振荡器引脚	4MHz	Out
4	VDD	工作电源电压	4.5-5.5V	In
5	TXD/38K	串行口发送/可以选择是否进行 38K 调制	0-5V	Out
6	RXD	串行口接受	0-5V	In
7	RST_	外部复位	0V	In
8	IRQ_	外部中断输入	0-5V	In
9	VPP	编程电压引脚	12V	In
10	EOSC1	时钟振荡器	32768Hz	In
11	EOSC2	时钟振荡器	32768Hz	Out
12	VDDA	调电时，RTC 的备用电源	3.3V	In
13-20	PC0-7	通用输入\输出端口 C	0-5V	In/Out
21-28	PB0-7	通用输入\输出端口 B	0-5V	In/Out

PLCC32 封装方式:

PIN NO	PIN NAME	FUNCTION	VALUE	IN/OUT
1	VSS	电源地	0V	In
2	XTAL1	振荡器引脚	4MHz	In
3	XTAL2	振荡器引脚	4MHz	Out
4	VDD	工作电源电压	4.5-5.5V	In
6	TXD/38K	串行口发送/可以选择是否进行 38K 调制	0-5V	Out

PIN NO	PIN NAME	FUNCTION	VALUE	IN/OUT
7	RXD	串行口接受	0-5V	In
8	RST_	外部复位	0V	In
9	IRQ_	外部中断输入	0-5V	In
10	VPP	编程电压引脚	12V	In
11	EOSC1	时钟振荡器	32768Hz	In
12	EOSC2	时钟振荡器	32768Hz	Out
13	VDDA	调电时，RTC 的备用电源	3.3V	In
15-22	PC0-7	通用输入\输出端口 C	0-5V	In/Out
23-27 30-32	PB0-7	通用输入\输出端口 B	0-5V	In/Out
28、29	PA0-1	通用输入\输出端口 A	0-5V	In/Out
5、14		空		

2.4.1 VDD 和 VSS

VDD 和 VSS 分别是工作电源和接地引脚。BL0938 采用单一的 5V 电源工作。当 MCU 引脚上发生快速变化的信号时，为了防止噪声问题，要在 MCU 旁边安置电容。可以象图 1-2 那样在尽可能靠近 MCU 的地方，放置一个旁路电容。C2 的电容容量可根据需要由用户确定，它可使引脚与地之间具有大电流通过能力。

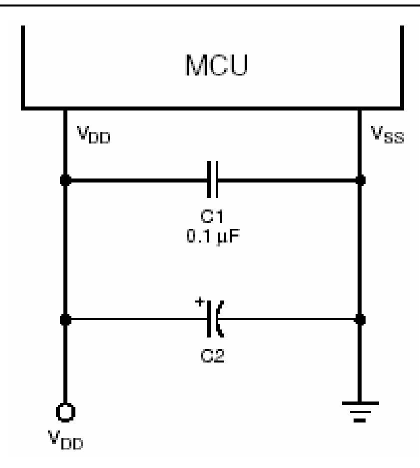


图 1-2 旁路电容

2.4.2 OSC1 和 OSC2

OSC1 和 OSC2 引脚用来控制连接在片振荡器。根据掩膜时的选择，振荡器可以选择以下几种：

- 晶体
- 阻容网络
- 外部时钟信号

内部振荡器频率是 f_{osc}。MCU 将内部振荡器的输出频率进行 2 分频产生内部时钟频率 f_{op}。

2.4.3 晶体振荡器

在 OSC1 与 OSC2 引脚之间连接一个晶体振荡器可以驱动片内振荡器。图 1－2 给出了典型的 AT 切割、电阻并联的晶体振荡器电路连接方式。参照晶振供应商的建议选用合适的晶振参数，因为它决定了外接部件提供的起振的可靠性和振荡的稳定性。振荡电路中选择电容容量时应当考虑分布电容的影响，为了减小输出失真，晶振和电容应当安装在尽可能靠近 MCU 引脚的地方。

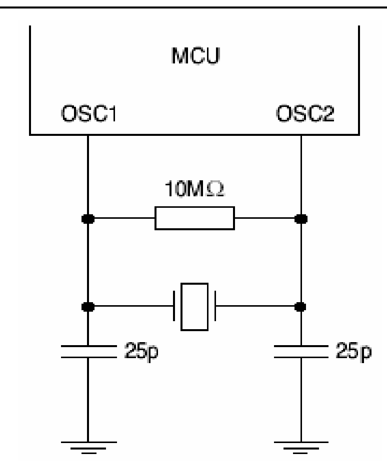


图 1-2 有反馈电阻的晶振连接

2.4.4RC 振荡器

当振荡频率要求不高，为了最大限度地降低成本，掩膜时可以选择 RC 振荡器，其应用有两种方式： 一是将 OSC1 和 OSC2 连接。频率约为 2MHZ。二是 OSC2 和 VDD 之间串入电阻。电阻值为 30K 到 110K 变化时，频率约从 4MHZ 到 1.5MHZ 变化。供用户参考。

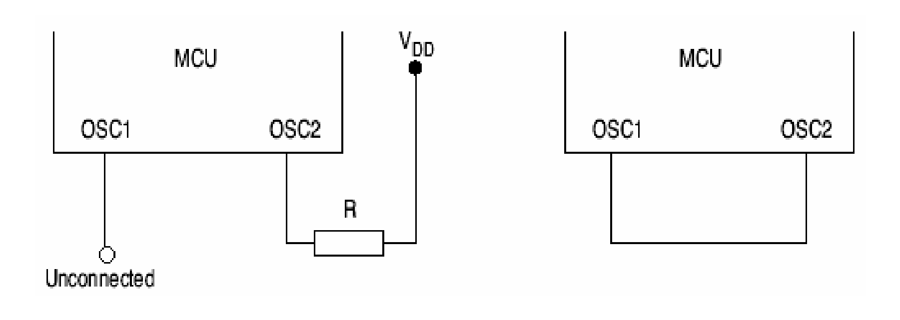
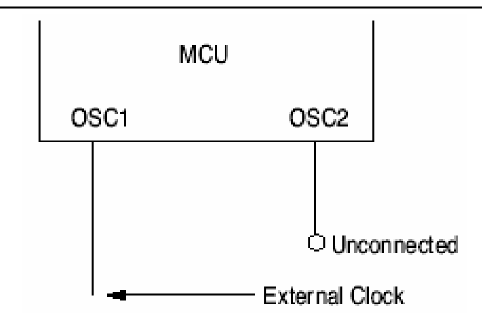


图 1-3 RC 振荡器的连接

2.4.5 外部时钟

在掩膜选用 RC 振荡器时，可以在 OSC1 脚输入其他兼容的 CMOS 器件产生的时钟信号，如图 1-4 所示，可以让 OSC2 脚悬空。



2.4.6 复位脚($\overline{\text{RESET}}$)

$\overline{\text{RESET}}$脚上的逻辑零可以强制 MCU 进入复位状态。

2.4.7 中断请求（$\overline{\text{IRQ}}$）

在 BL0938 中，中断系统所涉及的中断主要是定时器中断、软件中断、外部中断、RTC 中断、UART 中断、低电压检测中断、键盘中断。中断请求脚是异步外部中断脚。

2.4.8 PA1-PA0

A,B,C 端口是 8bit 双向口，端口上的所有腿可由相对应的数据方向寄存器和数据寄存器来控制。

I/O PIN 功能

R/W	DDR	I/O PIN 功能
0	0	引腿为输入，内部数据写入数据寄存器。
0	1	引腿为输出，内部数据写入数据寄存器，并输出。
1	0	引腿为输入，读入引腿状态。
1	1	引腿为输出，读数据寄存器内容。

当杂用控制寄存器（MCR）中的 KBIE 置 1 时，PA7－PA0 可用键盘中断。在键盘中断屏蔽寄存器（KIMR）中，通过设置对应位，可对 PA 口的个别管脚键盘设置键中断输入。

2.4.9 PB7－PB0

PB7-PB0 是通用双向 I/0 口，B 口的 8 个引脚。

2.4.10 PC7－PC0

PC7-PC0 是通用双向 I/0 口，B 口的 8 个引脚。

2.5 功能详述

2.5.1 存 贮 器

BL0938 片内存贮器的结构

地址	区域
$0000 $0010	用户寄存器 16字节
$000F $008F	RAM 128字节
$0090 $009E	用户寄存器 15字节
$009F $00BF	未用
$00C0 $00FF	堆栈 64字节
$0100 $0FFF	未用
$1000 $1EFF	用户ROM 3840字节
$1F00 $1FEF	自检ROM 240字节
$1FF0 $1FFF	用户矢量 16字节

寄存器	地址
A口数据寄存器	$00
B口数据寄存器	$01
C口数据寄存器	$02
A口方向寄存器	$04
B口方向寄存器	$05
C口方向寄存器	$06
定时器数据寄存器TDR	$08
定时器控制寄存器TCR	$09
端口选择寄存器	$0A
键盘中断屏蔽寄存器	$0B
杂用控制寄存器MCR	$0C
特殊控制寄存器	$0D

寄存器	地址
UART控制寄存器URC	$90
UART状态寄存器USR	$91
URT波特率寄存器UBR	$92
URT接收寄存器URR	$93
URT发送寄存器UTR	$94
URT中断控制寄存器UIR	$95
RTC控制寄存器RCR	$98
RTC时钟校准寄存器RAR	$99
RTC秒寄存器RSR	$9A
RTC分钟寄存器RMR	$9B
RTC小时寄存器RHR	$9C
RTC星期寄存器RWR	$9D
RTC星期数寄存器RDR	$9E

矢量	地址
RTC时钟校中断矢量	$1FF0
UART中断矢量	$1FF2
键盘中断矢量	$1FF4
计时器中断矢量	$1FF6
外部中断矢量	$1FF8
低电压中断矢量	$1FFA
软件中断矢量	$1FFC
复位	$1FFE

图 2-1 存贮器结构图

如图 2-1 所示 CPU 可以 寻址 8K 字节 的存储空间。 存贮器的 ROM 区 用来存程 序指令、固定数据、用户定义矢量和中断服务程

序，存贮器的 RAM 区用来放置可变数据。用户寄存器映射在存贮器中，这样 CPU 可以用与访问存贮器其他区域相同的寻址方式来寻址 用户寄存器空间。

2.5.2 外围接口用户存贮器

存贮器的前 16 字节地址即$0001－$000F 以及存贮器地址$0090－$009E 的 15 个字节，是用户寄存器区。它们分别为数据寄存器，方向寄存器，定时数据寄存器，定时控制寄存器，I/O 端口上拉电阻选择寄存器，键盘中断屏蔽寄存器，杂用控制寄存器，特殊控制寄存器，UART 控制寄存器 URC，UART 状态寄存器 USR，URT 波特率寄存器 UBR，URT 接收寄存器 URR，URT 发送寄存器 UTR，URT 中断控制寄存器 UIR，RTC 控制寄存器 RCR，RTC 时钟校准寄存器 RAR，RTC 秒寄存器 RSR，RTC 分钟寄存器 RMR，RTC 小时寄存器 RHR，RTC 星期寄存器 RWR，RTC 星期数寄存器 RDR。

请参阅图 2-2。

	Bit 7	**6**	**5**	**4**	**3**	**2**	**1**	**Bit 0**	
$0000	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	PORTA
$0001	Pb7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	PORTB
$0002	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	PORTC
$0003	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0	PORTD
$0004	DDRA 7	DDRA 6	DDRA 5	DDRA 4	DDRA 3	DDRA 2	DDRA1	DDRA 0	DDRA
$0005	DDRB 7	DDRB 6	DDRB 5	DDRB 4	DDRB 3	DDRB 2	DDRB1	DDRB 0	DDRB
$0006	DDRD 7	DDRD 6	DDRD 5	DDRD 4	DDRD 3	DDRD 2	DDRD1	DDRD 0	DDRC
$0007	DDRD 7	DDRD 6	DDRD 5	DDRD 4	DDRD 3	DDRD 2	DDRD1	DDRD 0	DDRD
$0008	TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0	TDR
$0009	TIF	TIM	TCEX	TINE	PRER	PRE2	PRE1	PRE0	TCR
$000A	PFS	COPE	PIL	PDP	PCP	PBP	PB1	PB0	POPR
$000B							KBE1	KBE0	KBIM

	Bit 7	6	5	4	3	2	1	Bit 0	
$000C	KBIE	0/KBIC	INTO	INTE		SM			MCR
$000D			LVFG	E38K	E32K	RTCS	LVIF	LVIE	SCR
$0090				OPEN	EPEN	UART EN	INLS1	INLS0	UCR
$0091	FREF	PAEF	OVEF	TSEF			RDAF	TREF	USR
$0092	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0	UBR
$0093	R7	R6	R5	R4	R3	R2	R1	R0	URR
$0094	T7	T6	T5	T4	T3	T2	T1	T0	UTR
$0095							TDEE	RDFE	UIR
$0098	RTCIF	RTCIE		ADJ/X STP	Flag	CLEN	CT1	CT0	RCR
$0099	XSL	F6	F5	F4	F3	F2	F1	F0	RAR
$009A		SD6	SD5	SD4	SD3	SD2	SD1	SD0	RSR
$009B		MD6	MD5	MD4	MD3	MD2	MD1	MD0	RMR
$009C			HD5	HD4	HD3	HD2	HD1	HD0	RHR
$009D						WD2	WD1	WD0	RWR
$009E	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0	RDR

图 2-2 I/O 寄存器 $0000-$009E

2.5.3 CPU 控制电路

控制电路是 BL0938 重要部件，整个 CPU 和 MCU 都是在控制电路的控制下正常工作。MCU 的所有同步信号，包括对寄存器的读写信号，包括对存储器的读写信号，对 A/DC 的控制信号，对中断系统的同步信号，ALU 的运算同步信号等，都由 CPU 控制电路产生和发出。

此外，控制电路还担负着对指令的计数，指令读出，指令的译码，指令的执行等工作。

2.5.4 ALU 部件

ALU 是 BL0938 的运算部件，可执行算术运算和逻辑运算。ALU 在本质上是一个加法器，因为四则运算在本质上可以用加法来实现。逻辑运算有逻辑加，逻辑乘，移位，按位加等。这些运算在加法器中通过一定的逻辑控制都能实现。

ALU 在执行算术或逻辑运算时会自动产生一些标志，如进位信号。

零标志等，它会送到条件码寄存器中保存。指令的执行，一般通过ALU 实现。

2.5.5 累加器(A)

累加器是一个 8 位寄存器，它与 ALU 共同实现累加功能。累加寄存器 A 主要功能是，运算前存放参与运算的一个操作数，和运算后存放运算结果。

累加寄存器 A 的输入端和 ALU 的输出相联，而输出端和内部总线及 ALU 的输出端相联。复位时，累加寄存器 A 中的内容不受影响。

2.5.6 变址寄存器 X（INDEX REGISTER）

变址寄存器 X 是一个 8 位寄存器，作用是指令变址。也可用作暂时寄存器。无偏移量变址寻址时，变址寄存器包括操作数的低位字节，高位字节的值为$00。在 8 位偏移量的变址寻址中，CPU 通过将变址寄存器内容与 8 位立即数相加而得到操作数地址。在 16 位偏移量的变址寻址中，CPU 通过将变址寄存器内容与 16 位立即数相加而得到操作数地址。

2.5.7 程序计数器 PC（PROGRAM COUNTER）

BL0938 得程序计数器是 16 位，用于对指令地址进行计数和存放。BL0938 最大寻址地址是$1FFF，所以在 16 位的程序计数器中，最高 3 位为无用位，CPU 中对其高 3 位的内容忽略，只把低 13 位作为程序计数器的真正内容。

2.5.8 堆栈指针 SP（STACK POINTER）

堆栈指针是一个 16 位的寄存器，在 BL0938 中的堆栈指针 SP 没有用到高 10 位，只用到低 6 位。在 16 位堆栈指针 SP 中，高 10 位固

定为“0000000011”，低 6 位的值指明栈顶。所以，16 位的堆栈指针 SP 寻址范围是 64 个存储单元，寻址范围从$00C0-$00FF.

在执行 MCU 复位或堆栈复位操作命令 RSP 时，堆栈指针就置为$00FF 即栈底。把数据压入堆栈，则 SP 减“1”，把数据从堆栈取出，则 SP 加“1”。无论是出栈还是入栈堆栈指针 SP 永远指向当前栈顶。

2.5.9 条件码寄存器 CCR（CONDITION CODE REGISTER）

条件码寄存器 CCR 是一个 8 位寄存器，它最高 3 位全是“1”，低 5 位则用于存放条件标志，分别为 H，I，N，Z，C。

（1）字节进位标志 H

H 是作加法时，在 ALU 的低 4 位向高位产生的进位标志。

（2）断屏蔽位 I

中断屏蔽位 I＝1 时，表示屏蔽内部和外部中断，此时如有中断请求产生，如果允许中断则需消存该信号。如中断屏蔽位 I＝0 时，则中断请求会被处理。

从中断返回指令 RTI 会把有关寄存器的内容从堆栈取出，并恢复在中断产生时的中断屏蔽位 I 的状态。在复位后，中断屏蔽位 I＝1，只有用清 I 位指令 CLI 和 STOP 或 WAIT 指令才能对其清零。

（3）负标志 N

负标志 N 用于反映刚执行的算术操作，逻辑操作，或者数据处理的结果是否为负。只要最高位 D7 为 1，则就把 N 置为“1”。

（4）零标志位 Z

零标志位 Z 用于反映刚执行的算术操作，逻辑操作，或者数据处理的结果是否为零。如果结果为“0”，则 Z＝1，否则 Z＝0。

（5）进位标志 C

进位标志 C 也称借位标志。用于反映刚执行的算术操作，逻辑操作，或者数据处理的结果，是否在累加器的最高位产生进位。在累加器的最高位产生进位，则 C＝1，否则 C＝0。进位标志 C 在移位和循环移位，转移，位试验指令时，也会受到影响，但此时并不表示执行的结果为“0”。

在执行加“1”指令 INC 或减“1”指令 DEC 时，并不影响进位标志 C。

进位标志 C 在补码运算时，反映最高的借位情况，也称借位标志。

2.6 中央处理单元(CPU)

2.6.1 CPU 寄存器

MCU5 中有 5 个寄存器。图 2－3 给出了 5 个 CPU 寄存器。CPU 寄存器不是存贮器的一部分。

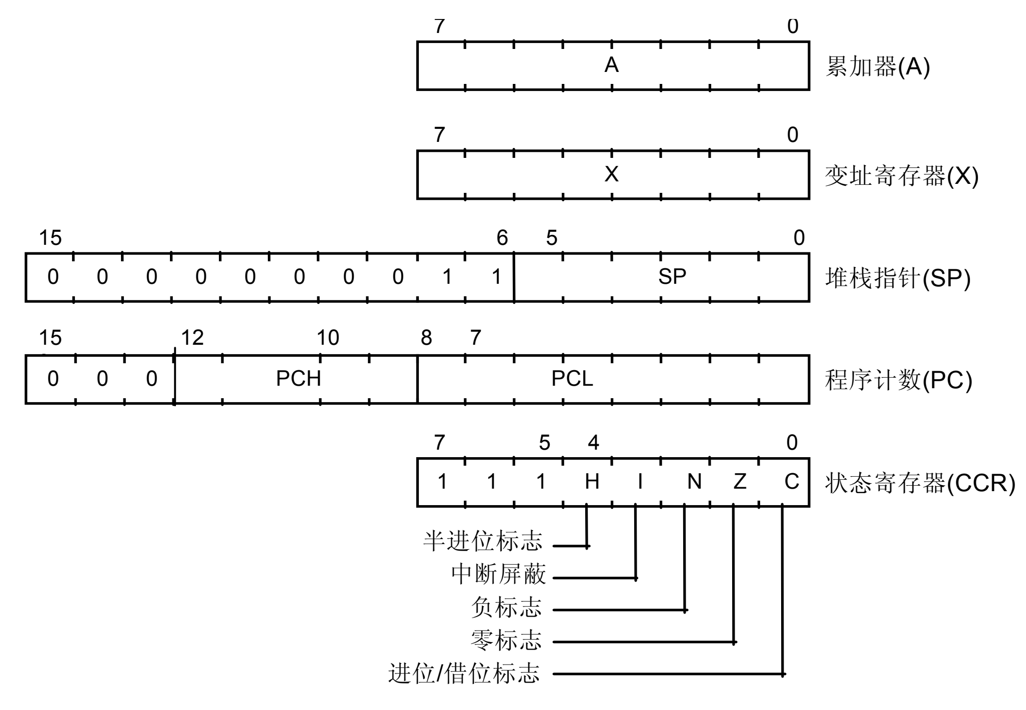


图 2－3. CPU 寄存器

2.6.2 累加器(A)

累加器是一个 8 位寄存器，它与 ALU 共同实现累加功能。累加寄存器 A 主要功能是，运算前存放参与运算的一个操作数，和运算后存放运算结果。累加寄存器 A 的输入端和 ALU 的输出相联，而输出端和内部总线及 ALU 的输出端相联。复位时，累加寄存器 A 中的内容不受影响。

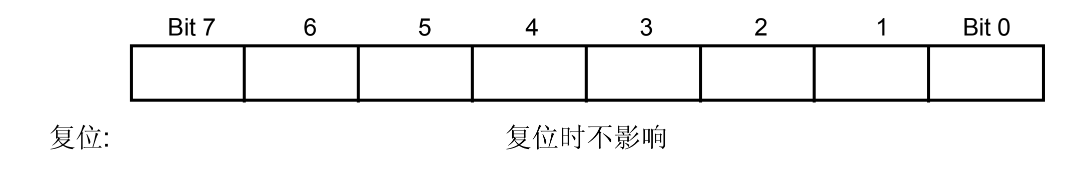


图 3-2. 累加器(A)

2.6.3 变址寄存器 X（INDEX REGISTER）

变址寄存器 X 是一个 8 位寄存器，它的作用是指令变址。有时，它也可用作暂时寄存器。变址寄存器 X 除了是一个专门用于变址的寄存器外，在结构上和一般的寄存器并无区别。

在 CPU 执行不同的变址方式时，其操作有区别。在无偏移量变址寻址时，变址寄存器包括操作数的低位字节，高位字节的值为$00。在 8 位偏移量的变址寻址中，CPU 通过将变址寄存器内容与 8 位立即数相加而得到操作数地址。在 16 位偏移量的变址寻址中，CPU 通过将变址寄存器内容与 16 位立即数相加而得到操作数地址。复位时，累加寄存器 A 中的内容不受影响。

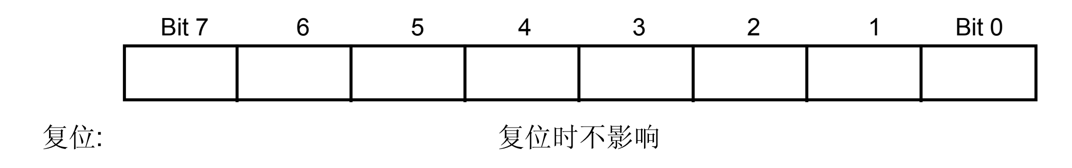


图 2-5. 变址寄存器(X)

2.6.4 堆栈指针 SP（STACK POINTER）

堆栈指针是一个 16 位的寄存器，在 BL0938 中的堆栈指针 SP 没有用到高 10 位，只用到低 6 位。在 16 位堆栈指针 SP 中，高 10 位固定为“0000000011”，低 6 位的值指明栈顶。所以，16 位的堆栈指针 SP 寻址范围是 64 个存储单元，寻址范围从$00C0-$00FF.

在执行 MCU 复位或堆栈复位操作命令 RSP 时，堆栈指针就置为 $00FF 即栈底。把数据压入堆栈，则 SP 减“1”，把数据从堆栈取出，则 SP 加“1”。无论是出栈还是入栈堆栈指针 SP 永远指向当前栈顶。

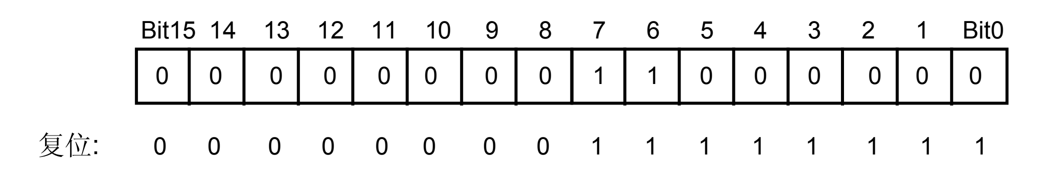


图 2-6. 堆栈指针 (SP)

如果子程序或中断处理程序使用的堆栈空间超过 64 个字节，堆栈指针循环置为$00FF，从而用新的数据将前面存贮的数据覆盖掉，这可能引起系统工作的崩溃调用一次子程序使用 2 个堆栈单元，而每一级中断要使用 5 个堆栈单元。

2.6.5 程序计数器 PC（PROGRAM COUNTER）

BL0938 得程序计数器是 16 位，用于对指令地址进行计数和存放。BL0938 最大寻址地址是$1FFF，所以在 16 位的程序计数器中，最高 3 位为无用位，CPU 中对其高 3 位的内容忽略，只把低 13 位作为程序计数器的真正内容。

在取指令时，程序计数器在读出每一字节时，就自动执行加 1 操作，从而自动指向下一个字节的地址。如果读取存储器中的数据，则程序计数器 PC 给出的是数据地址。在执行转移指令 jump，branch 时，

PC 中存放转移地址。在执行中断时，则 PC 中存放中断人口地址，从而指向中断人口。

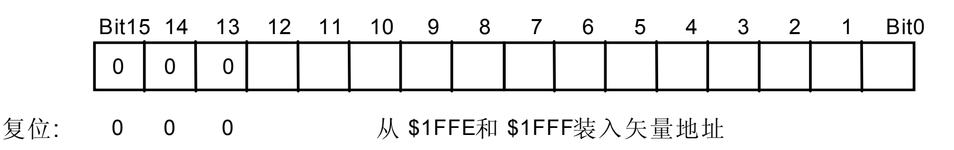


图 3-5.程序计数器(PC)

2.6.6 条件码寄存器 CCR（CONDITION CODE REGISTER）

条件码寄存器 CCR 是一个 8 位寄存器，它最高 3 位全是“1”，低 5 位则用于存放条件标志，分别为 H，I，N，Z，C。

（6）字节进位标志 H

H 是作加法时，在 ALU 的低 4 位向高位产生的进位标志。

（7）断屏蔽位 I

中断屏蔽位 I＝1 时，表示屏蔽内部和外部中断，此时如有中断请求产生，如果允许中断则需消存该信号。如中断屏蔽位 I＝0 时，则中断请求会被处理。

从中断返回指令 RTI 会把有关寄存器的内容从堆栈取出，并恢复在中断产生时的中断屏蔽位 I 的状态。在复位后，中断屏蔽位 I＝1，只有用清 I 位指令 CLI 和 STOP 或 WAIT 指令才能对其清零。

（8）负标志 N

负标志 N 用于反映刚执行的算术操作，逻辑操作，或者数据处理的结果是否为负。只要最高位 D7 为 1，则就把 N 置为“1”。

（9）零标志位 Z

零标志位 Z 用于反映刚执行的算术操作，逻辑操作，或者数据处理的结果是否为零。如果结果为“0”，则 Z＝1，否则 Z＝0。

（10）　进位标志 C

进位标志 C 也称借位标志。用于反映刚执行的算术操作，逻辑操作，或者数据处理的结果，是否在累加器的最高位产生进位。在累加器的最高位产生进位，则 C＝1，否则 C＝0。进位标志 C 在移位和循环移位，转移，位试验指令时，也会受到影响，但此时并不表示执行的结果为“0”。

在执行加“1”指令 INC 或减“1”指令 DEC 时，并不影响进位标志 C。

进位标志 C 在补码运算时，反映最高的借位情况，也称借位标志。

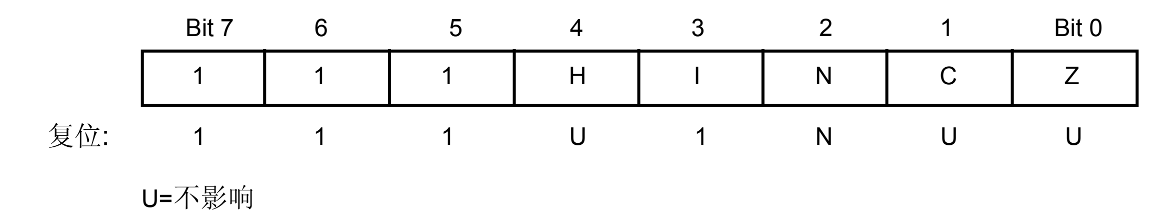


图 3-6.状态寄存器(CCR)

2.7 中断系统

中断系统是 BL0938 用于实时处理和实时控制的重要部件，在 BL0938 中共有 8 种不同的中断：定时器中断、软件中断、外中断、RTC 中断、UART 中断、低电压检测中断、键盘中断和复位。

2.7.1 中断请求的产生

BL0938 中断系统主要是定时器中断、外中断、RTC 中断、UART 中断、低电压检测中断、键盘中断。至于复位和软件中断，它们是不受中断屏蔽位控制的，即是不可屏蔽的。

2.7.2 中断的有关寄存器

中断的有关寄存器含有三种：一种是对中断系统的任何中断请求

进行屏蔽或允许控制的寄存器；一种是对某种中断请求进行允许或屏蔽控制的寄存器；一种是对同一种中断的多个输入端进行选择的控制寄存器。

2.7.3 中断系统屏蔽有关的寄存器

可对中断系统进行允许或屏蔽的寄存器是条件码寄存器 CCR。条件码寄存器是一个 8 位的寄存器，它的 D3 位称为 I 标志，这个标志就是用于控制中断的，I 标志在条件码寄存器 CCR 中的位置如前面的图。

当 I＝1 时，则屏蔽中断。当 I＝0 时，则允许中断。

对 CCR 中的 I 标志置“1”的条件有如下几种：

（1）响应中断时，则 CPU 把寄存器内容压入堆栈后，而在取出中断向量之前，自动把 I 标志置“1”。

（2）在执行复位时，对 I 标志置“1”。

（3）在 I 标志为“1”时，执行软件中断指令并进入中断处理，则从中断返回指令 RTI 返回主程序，I 标志为“1”。

（4）有置 I 标志指令 SEI，对 I 标志置 1。

对 CCR 中的 I 标志清“0”的条件有如下几种：

（1）有清 I 标志指令 CLI，对 I 标志清零。

（2）有 STOP 指令把 BL0938 置为停止状态时，会把 I 标志清零。

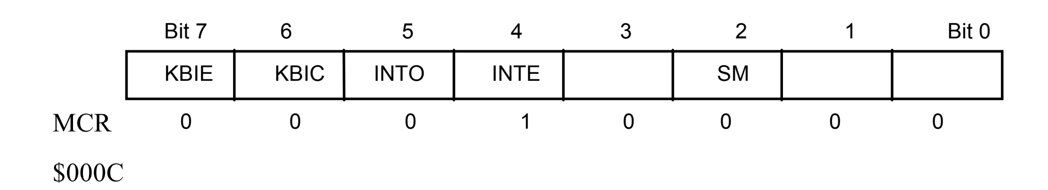
（3）用 WAIT 指令把 BL0938 置为等待状态时，会把 I 标志清零。

2.7.4 对各种中断分别进行控制的寄存器

杂用控制寄存器 MCR 是可以对各种中断请求进行控制的寄存器，它是一个 8 位的寄存器，它的主要作用是对键盘中断以及外部中断进

行分别的控制。其它作用是对振荡器的分频率进行控制。

杂用控制寄存器 MCR 各位意义如图：



	Bit 7	6	5	4	3	2	1	Bit 0
	KBIE	KBIC	INTO	INTE		SM		
MCR $000C	0	0	0	1	0	0	0	0

杂用控制寄存器 MCR

杂用控制寄存器 MCR 中各位的符号及意义：

（1）D7：KBIE＝1 允许键盘中断，KBIE＝0 屏蔽键盘中断。

（2）D6：KBIC＝1 清键盘中断的锁存信号，KBIC＝0 保留键盘中断的锁存信号。

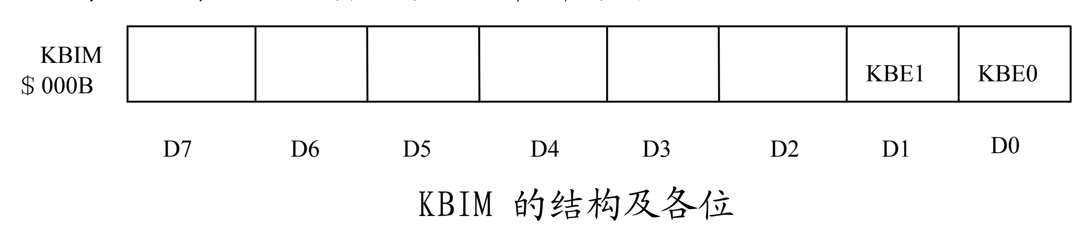
（3）D5：INTO＝0 允许外部中断信号 IRQ 为电平信号或边沿信号。INTO＝1 只允许外部中断信号 IRQ 为边沿信号。

（4）D4：INTE＝1 允许 IRQ 中断。INTE＝0 屏蔽 IRQ 中断。

（5）D2：SM＝1 在振荡器和内部时钟驱动器之间加一个系数位 16 的分频器，降低系统的时钟频率。SM＝0 则不接分频器。SM 可通过掩模选择来决定。

2.7.5 中断输入端选择控制寄存器

在 BL0938 键盘中断有两个输入端，对键盘中断的输入端选择控制是有键盘中断屏敝寄存器 KBIM 完成的。KBIM 是一个 2 位的寄存器，每位用于 PA 口对应的键盘中断的输入。



KBIM $ 000B							KBE1	KBE0
	D7	D6	D5	D4	D3	D2	D1	D0

KBIM 的结构及各位

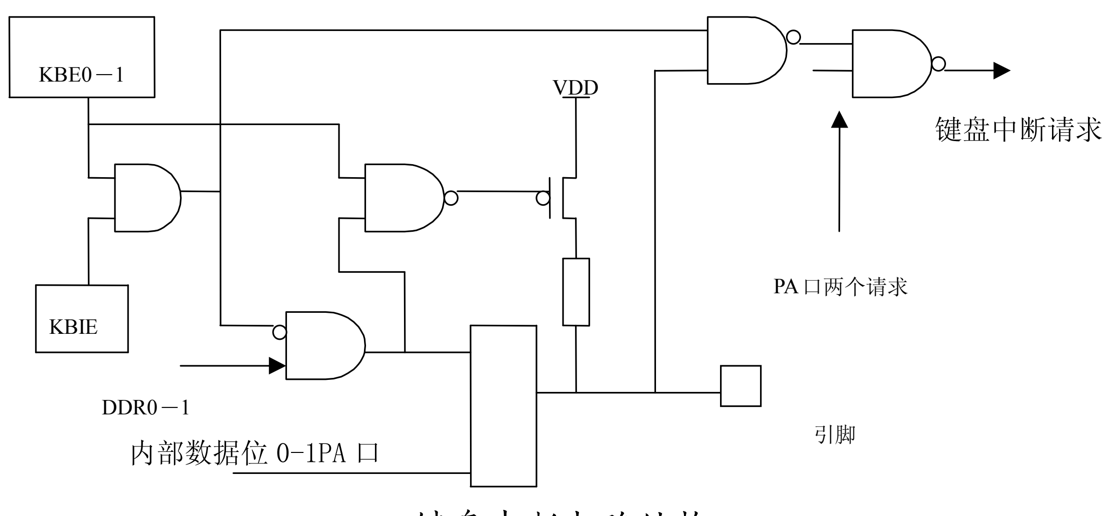
KBEi = 1，则选择 PAi 为键盘中断输入端，并且该脚接 20K 上拉电阻。

KBEi = 0，则 PAi 不作为键盘中断输入端，仅为一般的 I/O 引脚。

在接电源或复位时，KBIM 被清零。

2.7.6 键盘中断请求的产生

键盘中断共有 8 个输入端，电路结构如图。



键盘中断电路结构

2.7.7 定时器中断请求的产生

定时器中断主要有如下几个条件，它们相结合一起作用才能产生定时器中断请求。

(1) 定时器中断屏敝标志 TIM

TIM = 1，则屏敝定时器中断，在 TIM = 0，允许定时器中断。

(2) 定时器中断请求标志 TIF

TIF = 1，表示有定时器中断请求，TIF = 0，没有定时器中断请求。

(3) 定时器数据寄存器

定时器数据寄存器 TDR 在进行减法计数时的结果为零时，则会对定时器中断请求标志 TIF 置“1”，表示有定时器中断请求产生。

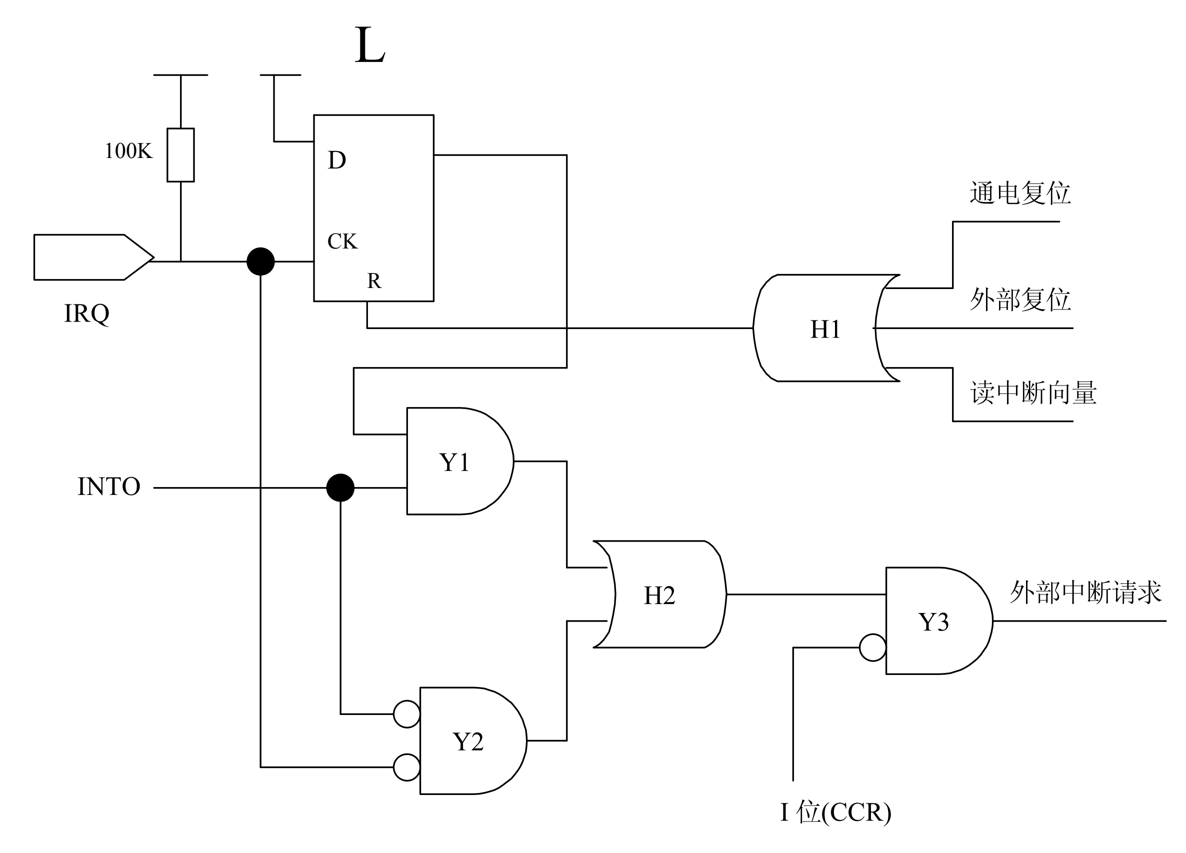
定时器控制寄存器 TDR 中含有定时器中断屏敝标志 TIM 和定时器中断请求标志 TIF。

当定时器中断屏敝标志 TIM＝0 时，一旦定时器数据寄存器计数结果为零，那么就产生定时器中断请求，即令 TIF＝0。在定时器电路中，定时器中断请求是由溢出检测电路，中断电路产生的。

2.7.8 外部中断请求的产生

在 BL0938 中，外部中断有 1 个，是 IRQ 中断。以 IRQ 说明外部中断产生的产生过程。

外部中断 IRQ 的电路框图，它由一个锁存器和有关的门电路组成。



外部 IRQ 中断电路

外部 IRQ 中断的 产生条件有如下几个:

(1) 条件码寄存器 CCR 的中断屏敝标志位 I 的 1 状态，I＝1，屏蔽

中断；I = 0，允许中断。

(2) 杂用控制寄存器 MCR 中的外部中断信号选择标志 INT0 的状态。在 INT0 = 0 时，外部中断信号可以为电平信号或边沿信号；在 INT0 = 1 时，外部中断信号只能是边沿信号。

(3) 外部中断信号 IRQ 必需和 INT0 状态所约束的状态一致。即 INT0 = 0 时，IRQ 可以是低电平，或由高电平变成低电平的边沿信号。在 INT0 = 1 时，IRQ 只能是由高电平变成低电平的边沿信号。

2.7.9 实时时钟中断请求的产生

实时时钟中断主要有如下几个条件，它们相结合一起作用才能产生实时时钟中断请求。

(1) 实时时钟中断使能位 RTCIE

当 RTCIE=1，不产生 RTC 中断；当 RTCIE=0，产生 RTC 中断信号，中断信号根据 CT1、CT0 的编码不同输出不同的周期性中断。

(2) 清实时时钟中断标志 RTCIF

RTC 产生中断后，中断信号将被锁存，当 RTCIF=1 时，清 RTC 中断锁存信号；RTCIF=0 时，RTC 中断锁存信号不变。在响应 RTC 中断之后，在中断处理子程序中应对 RTCIF 置位，否则芯片将不断响应 RTC 中断。在读 RTCIF 位时将始终读出为“0”。

(3) 中断周期选择标志：CT1、CT0。

用于选择 RTC 中断的周期，缺省值为 00。

00：产生周期为 1 秒的中断信号；

01：产生周期为 256 星期的中断信号；

10：产生周期为 10 秒的中断信号；

11：产生周期为 60 秒的中断信号。

(4) 实时时钟控制寄存器 RCR

实时时钟控制寄存器是一个用于控制 RTC 模块工作方式的寄存器，它的主要功能是： 秒信号的四舍五入进位功能，时钟的掉电检测，读写控制标志位，以及可编程的中断源选择等功能。

2.7.10 UART 中断请求的产生

UART 中断主要有如下几个条件，它们相结合一起作用才能产生 UART 中断请求。

(1) 发送寄存器空中断使能 TDEE

TDEE = 1 产生发送寄存器空中断，TDEE = 0 屏蔽发送寄存器空中断。若 TDEE 位和 TREF 位同时为 1 时，产生中断。

(2) 接收寄存器满中断使能 RDFE

RDFE = 1 产生接收寄存器满中断，RDFE = 0 屏蔽接收寄存器满中断。若 RDFE 位和 RDAF 位同时为 1 时，产生中断。

(3) UART 状态寄存器 USR

Bit7(FREF)：帧结构出错标志位， 在接收数据时，波特率出错、停止位出错、对一位数据两次采样的结果不等都能该位置位。（对起始位，如果两次采样的结果不等，就认为该位不是起始位，不使 FREF 置位；对数据位和校验位，如果两次采样的结果不等，使 FREF 置位；对于停止位，如果两次采样的结果不等，使 FREF 置位，如果两次采样的结果相等但不是 1，也会使 FREF 置位。）

Bit6(PAEF)： 奇偶校验出错标志位，在接收数据时，奇偶校验

位出错时该位置位。

Bit5(OVEF)：溢出标志位，接收寄存器中有数据，接收移位寄存器中又有数据送入接收寄存器时该位被置位，原接收寄存器中的数据被新数据覆盖。

Bit4(TSEF)：发送移位寄存器空标志位，当发送移位寄存器空时该位被置位。

Bit1(RDAF)：接收寄存器满标志位，当数据接收完后被送入接收寄存器后该位置位，读接收寄存器之后该位被清零。

Bit0(TREF)：发送寄存器空标志位，当发送寄存器可以被写入数据时，该位被置位。

对 USR 进行读操作之后,USR 的 Bit7 至 Bit5 位清零;Bit4、Bit1 与 Bit0 根据实际的情况由硬件改变。对 UCR 进行写操作之后，USR 的所有位被清零。

2.7.11 低电压中断请求的产生

低电压中断主要有如下几个条件，它们相结合一起作用才能产生低电压中断请求。

(1) 低电压中断使能 LVIE

LVIE=1 时允许低电压中断；LVIE=0 时不允许低电压中断。

(2) 清低电压中断位 LVIF

当产生低电压中断后，中断信号将被锁存，LVIF=1 时为清中断锁存信号；LVIF=0 时不清中断锁存信号，在响应低电压中断后，在中断处理子程序中应对 LVIF 位置位，否则芯片将不断响应低电压中断，读 LVIF 的内容时将始终读出“0”。

(3) 低电压标志位 LVFG

它是一位只读标志位，它是用来指示当前电压状况的，LVFG=1 表示电源电压低于 4.3V; LVFG=0 表示电源电压高于 4.3V。低电压中断模块的作用是在掉电前产生中断信号，芯片通过响应中断来完成重要的数据传送和保存，以防止由于掉电引起的数据丢失，低电压中断是采用边沿触发的方式，电源电压降低之后将产生中断信号，响应中断并清中断锁存标志位之后，若此时电源电压仍维持在低电压状态但又不是掉电时，芯片不会再次发出中断信号，此时要判断电源电压状态可通过读出 LVFG 的值来实现。

2.7.12 中断的响应过程

当发生中断时 BL0938 的 CPU 将寄存器内容保存至堆栈，对中断屏蔽位置“1”以防止其它中断。与复位不同，硬件中断不停止现行指令的执行，相反，暂悬发生的中断直到执行完现行的指令为止。

如果中断没有屏蔽(CCR 的 I 位清零)并且对应的中断允许位置“1”，则 CPU 执行中断过程。否则，它取出程序的下一条指令并继续执行。如果发生中断，CPU 执行完现行指令，把 CPU 寄存器状态压入堆栈。置 I 位“1”以防止发生新的中断。然后检查所有的暂悬的硬件中断，在 CPU 的寄存器入堆栈以后，若有多于一个的中断暂时悬而未执行，在中断向量存储区中具有最高向量地址的中断将首先得到处理。SWI 指令执行方法与其它指令相同，它于 I 位无关。中断响应后 CPU 从$1FF2-$1FFF 中取出对应的地址向量以执行中断服务程序。如 RTI 指令表示中断服务程序结束。它从堆栈中取出寄存器内容送回

CPU。并重新执行被中断了的程序。

RESET

置I位 — Y

低电压中断 — Y

IRQ中断 — Y — 清IRQ寄存器

TIMER中断 — Y

键盘中断 — Y

UART中断 — Y

实时时钟中断 — Y

取下一条指令

SWI中断 — Y

RTI中断 — Y

执行指令

PC->(SP, SP-1)
X->(SP-2)
A->(SP-3)
CC->(SP, 4)

对CCR的I位置1

送中断向量到PC

恢复

硬件中断处理流程

中断返回指令 RTI 使 CPU 通过退栈恢复寄存器的内容，如图所示。

退栈顺序	进栈顺序	堆栈	地址
			$00C0 (堆栈底部)
			$00C1
			$00C2
↓		. . .	. . .
5	1	状态寄存器	
4	2	累加器	
3	3	变址寄存器	
2	4	程序计数器 (高位)	
1	5	程序计数器 (低位)	
	↑	. . .	. . .
			$00FD
			$00FE
			$00FF (堆栈顶部)

图 4-3. 堆栈次序

表 4-1.复位/中断矢量地址

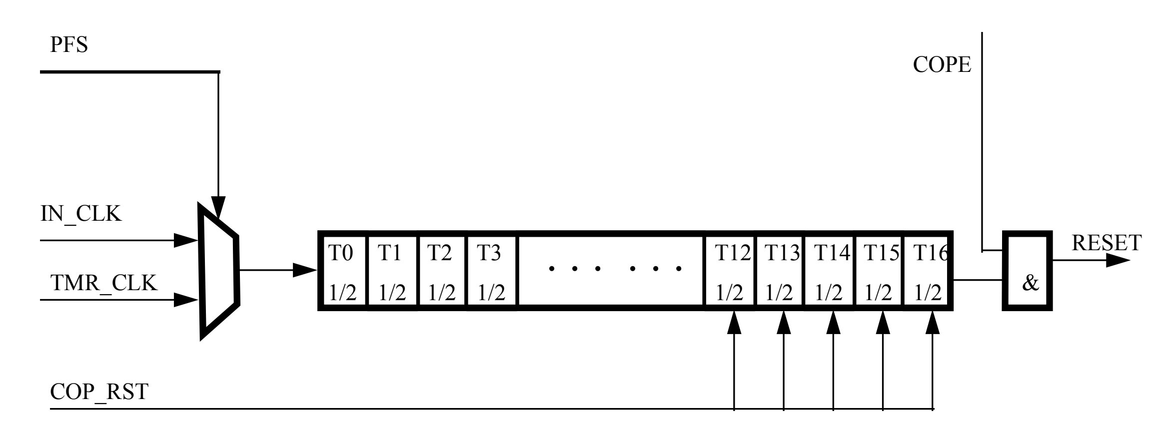
功能	源	相应的屏蔽位	总屏蔽位	优先级	矢量地址
复位	Power-On 复位脚 COP Watchdog[1] 非法地址	无	无	1	$1FFE-$1FFF
软件中断(SWI)	用户码	无	无	优先级与指令一致	$1FFC-$1FFD
低电压中断	VSS VDD	LVIE Bit LVIF Bit	I Bit	2	$1FFA-$1FFB
外部中断	$\overline{IRQ}$ Pin	IRQE Bit	I Bit	3	$1FF8-$1FF9
定时器中断	TOF Bit RTIF Bit	TOFE Bit RTIE Bit	I Bit	4	$17F7-$17F6
键盘中断	PA1 Pin[2] PA0 Pin[2]	KBEi Bit	I Bit	5	$17F5-$17F4
UART 中断	TxD Pin RxD Pin	TDEE Bit RDFE Bit	I Bit	6	$17F3-$17F2
RTC 时钟中断	RTCIF Bit	RTCIE Bit	I Bit	7	$17F1-$17F0

2.8 COP 复位 – WATCHDOG 说明

1、简介

当软件发生错误时，COP WATCHDOG 就会使 MCU 复位。正确运行的软件会周期复位 WATCHDOG 而防止其计数溢出而复位 MCU。COP WATCHDOG 的功能是可编程选择的。

2、COP Watchdog 框图



3、使用

1)掩膜选择

通过掩膜选择可以设置 WATCHDOG 自 POWER－ON 开始就始终起作用或是可编程的。

（A）当掩膜选择为“1”时，WATCHDOG 自 POWER－ON 起就始终起作用；

（B)当掩膜选择为“0”时,WATCHDOG 是否起作用是软件可编程选择的。

2)状态的设置

地址为$000A 的寄存器 POPR 的最高两位可以选择 WATCHDOG 的输入时种信号和 WATCHDOG 是否起作用。

POPR

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	reset

$000A	PFS	COPE	PIL	PDP	PCP	PBP	PB1	PB0	$00

PFS------Pre_Frequency_Select

1（set）------Watchdog 的输入时钟信号为 TIMER 中 PRESCLR 分频输出信号 TMR_CLK;

0（clear）------Watchdog 的输入时钟信号为内部总线时钟 IN_CLK，频率为 2MHZ，此时 Watchdog 的溢出时间为 64mS。

PFS＝“1”时 Watchdog 的溢出时间

PRE2	PRE1	PRE0	分频率	溢出时间 （mS）
0	0	0	1	64
0	0	1	2	128
0	1	0	4	256
0	1	1	8	512
1	0	0	16	1024
1	0	1	32	2048
1	1	0	64	4096
1	1	1	128	8192

COPE------COP Enable

1（set）------Watchdog 开始工作;

0（clear）------Watchdog 不起作用。

3）复位 Watchdog

复位 Watchdog 的工作可以通过向地址$1FF0 写“1”来产生 COP_RST 信号完成，例如使用指令

```
LDA #$01
  STA $1FF0
```

2.9 定时器

定时器包括一个 8 位可编程定时器数据寄存器 TDR，一个 7 位可编程的预定标器。TDR 初值由程序设定然后递减至零。当计数器为零时，计数器中断标志位置“1”，即 TCR 第 7 位置“1”。如果定时器中断未被屏敝。即 TCR 的第六位以及 CCR 的 I 位为零，将产生中断。CPU 在执行完当前指令后，将有关寄存器压入堆栈然后从$1FF6，$1FF7 地址中取中断向量的开始中断服务过程。计数器不断进行循环计数，其值可由软件进行设定，而计数信号是内部时种还是外部信号，也由软件设定。CPU 可在任何时候对计数器进行读出，而不会影响其它的操作。计数器内容在读周期之前要稳定下来，并在读周期内不许变化。定时器中断标志始终保持“1”，除非软件对其清零，如果在定时器中断响应前对计数器进行写操作，此中断将会丢失，在非中断运行方式下 TCR 也可作状态查询用。预定标器是一个七位分频器，它可使定时输出时间延长。TCR$09 的第 0 位，第 1 位，第 2 位(PR0,PR1,PR2)可编程，以选最择合适的预定标器输出，用来作为计数器入。CPU 不能对预定标器进行直接读写操作，但对 TCR 第 3 位写“1”操作可以将预定标器内容清零，从而改变计数现行分频。

时种输入可构造为三种不同的输入方式，以及禁止方式，这些方式的确立由定时器输入允许标志 TCNE 和定时器外部时种标志 TCEX 以及 TCR 的第 3 和第 4 位决定。定时器的结构图含有定时器数据寄存器

TDR，计数器，预定标器选择逻辑，溢出检测电路，定时器控制寄存器 TCR，时种源逻辑等电路和部件。

定时器电路结构框图

Internal Bus
$08 TDR
Timer Data Register
TDR
Prescaler Select Logic
Interrupt
Overflow detect circuit
POR
$09 TCR
Timer Control Register
Reset
TIF TIM TCEX TINE PRER PR2 PR1 PR0
Clock Source Logic
External signal
Internal clock

2.9.1 时种源逻辑

时种源逻辑是一个定时器输入信号的一个选择电路，可对内部时种和外部输入计数信号进行选择。时种源逻辑是受定时器控制寄存器 TCR (Timer Control Register) 的 TCEX 标志和 TINE 标志控制的。TCEX 是定时器外部时种控制位，TCEX = 1 时，选择外部输入信号。TCEX = 0，选内部时种或内部时种和外部信号为定时器的输入信号。TINE 是定时器输入选择和允许位。它和 TCEX 位相结合，以确定是否

屏敝定时器输入，以何种方式输入定时器的计数信号。

TCEX，TINE 位及功能

TCEX	TINE	功能意义
0	0	内部时种作定时器的计数信号
0	1	内部时种和外部信号执行“与”，再去计数
1	0	屏敝定时器输入
1	1	外部信号作为定时器输入

2.9.2 定时器控制寄存器 TCR(Timer Control Register)

定时器控制寄存器 TCR 是一个用于控制定时器工作方式和工作状态的寄存器。它的主要功能是：对定时器的中断进行控制，对输入计数的时种源进行控制，对预定标器的处理能力进行控制。TCR 是一个 8 位寄存器，在寄存器中的地址为$0009,复位时它的内容为$44。它结构如图。

TCR $0009	TIF	TIM	TCEX	TINE	PRER	PRE2	PRE1	PRE0
	D7	D6	D5	D4	D3	D2	D1	D0

定时器控制寄存器 TCR

在 TCR 中，各位的意义如下:

D7: TIF 位是定时器中断请求标志。在定时器中断屏敝位 TIM 为 “0”时，定时器的计数器一旦计数结果为“0”，则把 TIF 置“1”，表示有定时器中断请求。在通电和复位时，本位为“0”。

D6: TIM 位，这是定时器中断屏敝位。在 TIM＝1 时，定时器计数结果为“0”也不会产生定时器中断请求。在 TIM＝0 时，定时器计数结果为“0”，则可产生定时器中断请求。在通电和复位时，会

把 TIM 位置“1”，从而屏敝定时器中断。要允许定时器中断，必需用软件把 TIM 清“0”。

D5：TCEX 位，定时器外部时种控制位。它和定时器输人选择和允许位 TINE 相结 合，以确定定时器输入信号的种类和方式。

D4：TINE 位，定时器输入选择和允许位。它和定时器外部时种控制位 TCEX 相结合，以确定定时器输入信号及输入方式。

D3：PRER 位，是预定标器的复位。当把 PERE 置“1”时，会把预定标器复位为“0”态。在复位时，不影响 PERE 的内容。不过，无论 PRER 的内容如何，对它执行读操作时，读的结果为“0”。

D2－D0：PRE2－PRE0，这 3 位是预定标器分频率选择位。在通电及复位时，这 3 位被置位“100”。即分频率为 16。如表

PRE2－PRE0 及分频率

PRE2	PRE1	PRE0	分频率
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

2.9.3 预定标器的分频器

分频器是一个 7 位分频器，它用于对定时器的输入计数信号执行

分频。分频器可以从 0－127 进行分频。它的分频结果送到预定标器的选择逻辑中进行处理。定时器控制寄存器 TCR 的预定标器复位位 PRER 为“1”时，则可对 分频器执行清零。

2.9.4 预定标器选择逻辑

预定标器选择逻辑是一个对分频器的结果执行选择的电路。分频器的计数结果送到预定标器选择逻辑，而定时器控制寄存器 TCR 的 PRE2－PRE0 位状态也送到预定标器选择逻辑。PRE2－PRE0 位状态使预定标器选择逻辑对分频结果进行相应的选择。由于分频数分别是 1，2，4，8，16，32，64，128，故而频率的变化按 1，1/2，1/4，1/8，1/16，1/32，1/64，1/128 变化。也即，输入信号频率是 f，那么预定标选择逻辑输出的信号频率则按分频率为 1，2，4，8，16，32，64，128 的情况分频。预定标器选择逻辑的输出送到定时器数据寄存器 TDR 中去进行减法计数。

2.9.5 定时器数据寄存器 TDR

定时器数据寄存器 TDR 是一个可以读写的 8 位寄存器，它的内容是 8 位的定时器计数器的现行值。在中断请求标志 TIF 置“1”时，说明 TDR 的内容减至“0”。定时器数据寄存器 TDR 是一个减法计数器，它接受来自预定标器选择逻辑的输出信号作为计数信号。每来一个计数信号，则定时器数据寄存器 TDR 的内容减“1”。在任何时候，可用软件读定时器数据寄存器 TDR 的内容，并且不影响 TDR 的正确计数过程。从 TDR 读取的内容中，可根据原来的 TDR 的内容进行比较，从而求出和原来的内容相比输入了多少内部时种数或外部信号数。

定时器数据寄存器 TDR 也可称为定时器计数器，它的地址为

$0008。在复位时，TDR 的内容被设置为全“1”，即$FF。

当定时器数据寄存器 TDR 执行减法计数，并当结果为 0 时，会产生溢出信号送到溢出检测电路及中断电路去。

2.9.6 溢出检测电路和中断电路是产生定时器中断请求的电路

溢出检测电路用于检测来自 TDR 的溢出信号，并把这个溢出信号送到中断电路。中断电路接收溢出检测电路的信号以及定时器控制寄存器 TCR 的定时器中断屏蔽位 TIM＝0 状态，从而产生中断请求。

2.10 低电压复位和低电压中断

低电压复位和低电压中断模块，用于配合用户的电源管理。这两个模块可通过码点选择来选择工作在 5V 电源系统中或 3.3V 电源系统中。

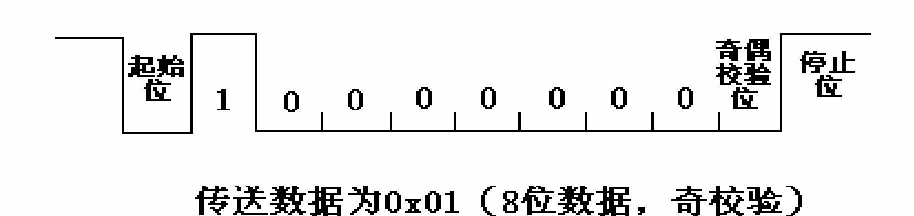
低电压复位模块的功能是在电压低于 3.0V(5V 电源系统)或 2.2V (3.3V 电源系统) 时产生复位信号，它工作与否是通过码点来选择，它的作用是在电源电压低于正常工作电压时使芯片复位，从而避免由于工作在低电压区时芯片工作状态异常引起的系统出错。

低电压中断模块的功能是在电压低于 4.5V(5V 电源系统)或 2.9V (3.3V 电源系统) 时产生中断信号，等待系统响应，它的中断向量入口地址为$1FFA，它的工作与 SCR ($0D) 寄存器中的 LVIE (D_0) 位 LVIF (D_1) 位 LVFG (D_5) 位有关。LVIE=1 时允许低电压中断；LVIE=0 时不允许低电压中断。LVIF 是清低电压中断位，当产生低电压中断后，中断信号将被锁存，LVIF=1 时为清中断锁存信号；LVIF=0 时不清中断锁存信号，在响应低电压中断后，在中断处理子程序中应对 LVIF 位置位，否则芯片将不断响应低电压中断，读 LVIF 的内容时将始终

读出“0”。LVFG 是低电压标志位，它是一位只读标志位，它是用来指示当前电压状况的，LVFG=1 表示电源电压低于 4.3V；LVFG=0 表示电源电压高于 4.3V。低电压中断模块的作用是在掉电前产生中断信号，芯片通过响应中断来完成重要的数据传送和保存，以防止由于掉电引起的数据丢失，低电压中断是采用边沿触发的方式，电源电压降低之后将产生中断信号，响应中断并清中断锁存标志位之后，若此时电源电压仍维持在低电压状态但又不是掉电时，芯片不会再次发出中断信号，此时要判断电源电压状态可通过读出 LVFG 的值来实现。

2.11 串行通讯模块 UART

UART 为一个半双工的串行异步通讯口，采用 1 位起始位，5—8 位的数据位，1 位奇偶校验位（此位也可为空奇偶位），1 位的停止位的数据格式，如下图所示。支持 600、1200、2400、4800、9600 五种传输波特率，支持数据位控制，帧错误判别，波特率控制，中断位查询，奇偶校验设置等功能。



传送数据为0x01（8位数据，奇校验）

2.11.1 控制寄存器（UCR 0x90）

Bit4(OPEN)：奇校验使能，1Enable，0 Disable。

Bit3(EPEN)：偶校验使能，1 Enable，0 Disable。

Bit2(UARTEN)：这位为 1 时，UART 才进行工作，复位后，此位为 0 建议每次初始化时，先设置 UBR，再设置 UCR。

Bit1 Bit0(INLS)：用于选择接收和发送的数据位长度。

INLS=0，8 位数据；

INLS=1，7 位数据；

INLS=2，6 位数据；

INLS=3，5 位数据。

对 UCR 进行写操作之后，UART 的状态将被复位，即所有的中断将被清除掉，接收或发送了一部分的数也将被清除掉；UCR 中的奇校验使能位和偶校验使能位不能同时为 1。

2.11.2 状态寄存器（USR 0x91）

Bit7(FREF)：帧结构出错标志位， 在接收数据时，波特率出错、停止位出错、对一位数据两次采样的结果不等都能该位置位。（对起始位，如果两次采样的结果不等，就认为该位不是起始位，不使 FREF 置位；对数据位和校验位，如果两次采样的结果不等，使 FREF 置位；对于停止位，如果两次采样的结果不等，使 FREF 置位，如果两次采样的结果相等但不是 1，也会使 FREF 置位。

Bit6(PAEF)：奇偶校验出错标志位，在接收数据时，奇偶校验位出错时该位置位。

Bit5(OVEF)：溢出标志位，接收寄存器中有数据，接收移位寄存器中又有数据送入接收寄存器时该位被置位，原接收寄存器中的数据被新数据覆盖。

Bit4(TSEF)：发送移位寄存器空标志位，当发送移位寄存器空时该位被置位。

Bit1(RDAF)：接收寄存器满标志位，当数据接收完后被送入接收寄存器后该位置位，读接收寄存器之后该位被清零。

Bit0(TREF)：发送寄存器空标志位，当发送寄存器可以被写入数据时，该位被置位。

对 USR 进行读操作之后，USR 的 Bit7 至 Bit5 位清零；Bit4、Bit1 与 Bit0 根据实际的情况由硬件改变。对 UCR 进行写操作之后，USR 的所有位被清零。

2.11.3 波特率寄存器（UBR 0x92）

波特率= 总线频率/[16*(UBR+1)]

上电复位值为 12

以 2MHz 总线频率为例，波特率与 UBR 的关系如下：

600 → 207

1200 → 103

2400 →51

4800 → 25

9600 → 12

由于写 UBR 会造成波特率的变化，若这种改写发生在数据传输中会引起异常出错，所以应在改写 UBR 寄存器后重送 UCR 的数据，使 UART 的工作状态复位。

2.11.4 接收寄存器（URR 0x93）

存放接收到的数据。若接收到的数据位小于 8 位，则将在 URR 的低位补零。

2.11.5 发送寄存器（UTR 0x94）

存放将发送的数据。若发送的数据位小于 8 位，发送时 UTR 的高位数据将被截断。

2.11.6 中断控制寄存器（UIR 0x95）

Bit1(TDEE)：发送寄存器空中断使能，1 Enable，0 Disable。若 TDEE 位和 TREF 位同时为 1 时，产生中断。

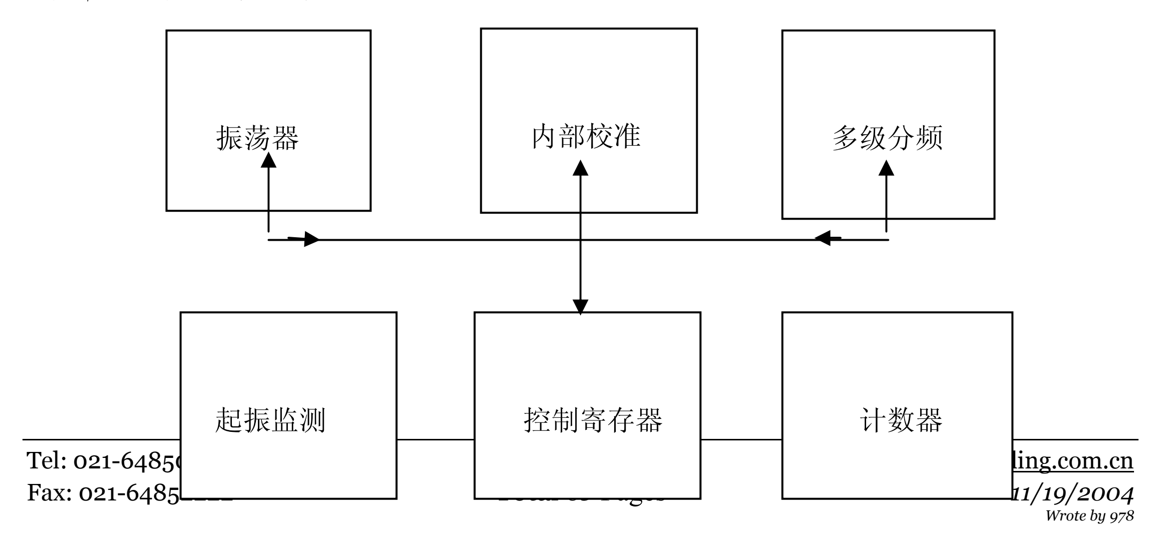
Bit0(RDFE)：接收寄存器满中断使能，1 Enable，0 Disable。若 RDFE 位和 RDAF 位同时为 1 时，产生中断。

注：在以上六个寄存器中，UCR、UBR、UTR、UIR 为只写寄存器，USR、URR 为只读寄存器；对 UCR、UBR、UTR、UIR 只支持按字节写操作，不支持按位写操作；对 USR、URR 同时支持按字节和位的读操作。UART 的中断入口地址为 1FF2。

2.12 实时时钟 RTC

RTC 为一个独立的时钟模块,它的计数信号来自 32.768KHz 的晶振,经过多级分频后计数,包括了秒、分钟、小时、星期、星期数等信息，提供 1 秒、10 秒、60 秒和 256 星期的四种中断选择，中断向量地址为$1FF0，

32.768KHz 和 32.000KHz 晶振的选择,以及内部时钟校正的功能。内部结构框图如下：



2.12.2 控制寄存器（RCR $98）

控制寄存器是一个用于控制 RTC 模块工作方式的寄存器,它的主要功能是：秒信号的四舍五入进位功能，时钟的掉电检测，读写控制标志位，以及可编程的中断源选择等功能。

	D7	D6	D5	D4	D3	D2	D1	D0
NAME	RTCIF	RTCIE	Reserved	Adj/Xstp	Flag	CLEN	CT1	CT0
Default	0	1		1	0	1	0	0
OP	W	W/R		W/R	R	W/R	W/R	W/R

1. RTCIF：清 RTC 中断标志。RTC 产生中断后，中断信号将被锁存，当 RTCIF=1 时，清 RTC 中断锁存信号；RTCIF=0 时，RTC 中断锁存信号不变。在响应 RTC 中断之后，在中断处理子程序中应对 RTCIF 置位，否则芯片将不断响应 RTC 中断。在读 RTCIF 位时将始终读出为“0”。
2. RTCIE：RTC 中断使能位。当 RTCIE=1，不产生 RTC 中断；当 RTCIE=0，产生 RTC 中断信号，中断信号根据 CT1、CT0 的编码不同输出不同的周期性中断。
3. ADJ/XSTP：这是一位复用位，在写入时是 ADJ（时钟校正位），在读出时是 XSTP（RTC 晶振停振检测位）。对 ADJ 置位后，如果秒级的计数大于 30，则分钟的计数加一，随后秒计数被清零，之后 ADJ 返回原状态。XSTP：只能读出，无写功能。时钟模块上电后为 1，对它实现初始化（写操作）后，变为 0。如果 XSTP

突然变为 1，说明在上一次读操作后，RTC 的 32KHz 晶振曾经停振，时间数据不对，提示更新时间数据。

4. FLAG：读写允许标志位。由于 RTC 中存在五级计数器，相互之间存在进位关系，若在进位发生时读写 RTC 的数据寄存器，有可能造成数据出错，所以设置该标志位，它在秒计数变化前的 1/128 秒变为 1，在秒的进位完成后立即恢复为 0。为了保证读写数据的正确，应在读写操作前检测该标志位是否为“0”，若为“0”则允许读写操作，否则禁止读写操作。同时为了读写的数据与期望值相符，建议对 RTC 数据的读写应成批进行。
5. CLEN：晶振测试信号输出允许位。CLEN=1，允许 32KHz 的信号输出；CLEN=0，不允许 32KHz 信号输出。该位缺省值为 1。(32KHz 信号是晶振的正弦波输出通过 RTC 内部反向器之后的方波信号，用于测试晶振的精度。)
6. CT1、CT0：中断周期选择。用于选择 RTC 中断的周期，缺省值为 00。

   00：产生周期为 1 秒的中断信号；

   01：产生周期为 256 星期的中断信号；

   10：产生周期为 10 秒的中断信号；

   11：产生周期为 60 秒的中断信号。

2.12.3 时钟校准寄存器（RAR $99）

该寄存器是用于对 32.000KHz 或 32.768KHz 晶振的选择和时钟校准。

	D7	D6	D5	D4	D3	D2	D1	D0

	D7	D6	D5	D4	D3	D2	D1	D0
W	XSL	F6	F5	F4	F3	F2	F1	F0
R	XSL	F6	F5	F4	F3	F2	F1	F0

1. 选用 32.768KHz 的晶体振荡器应设 XSL=0；选用 32.000KHz 的晶体振荡器应设 XSL=1。
2. 内部校准电路一旦启用，每隔 20 秒，根据用户的设定数据，自动对分频器的数据进行修改，从而达到调整频率的功能。一旦对该寄存器写入非零（(F6,F5,F4,F3,F2,F1,F0)不等于零）数据，内部校准电路将自动启用。当 F6=0 时，分频器的计数器要多计((F5,F4,F3,F2,F1,F0)-1)*2 个时钟才产生秒信号；当 F6=1 时，分频器要少计(!(F5,F4,F3,F2,F1,F0)+1)*2 个时钟就产生秒信号。
3. 当选用 32.768KHz 晶振时，而实际晶振频率是 32.770KHz，那么时钟走的比标准时钟快，为了让它和标准时钟同步，我们就应该在每个 20 秒使得分频器多计(32770-32768)*20 个时钟，也就是 40 个时钟周期，于是(F6,F5,F4,F3,F2,F1,F0)被设置成为(0,0,1,0,1,0,1)。这样，从一个较长的周期来看，相当于把 32.770KHz 的晶振等效为 32.768KHz 的晶振。

2.12.4 数据寄存器

1. RTC 中包括五个数据寄存器，分别为秒寄存器（RSR$9A）、分钟寄存器（RMR $9B）、小时寄存器（RHR $9C）、星期寄存器（RWR $9D）、星期数寄存器（RDR $9E）。
2. 以上五个寄存器除天数寄存器外全部采用 BCD 码计数，有效位

数分别为 7 位、7 位、6 位、3 位。星期数寄存器采用 16 进制方式计数，有效数据位 8 位。上述五个寄存器中无效的位读出的将是 0。

RSR		D6	D5	D4	D3	D2	D1	D0
RMR		D6	D5	D4	D3	D2	D1	D0
RHR			D5	D4	D3	D2	D1	D0
RWR						D2	D1	D0
RDR	D7	D6	D5	D4	D3	D2	D1	D0

小时计数采用 24 小时制的方式。

2.13 指令系统

本章介绍了 BL0938 的寻址方式和指令类型

2.13.1 寻址方式

BL0938CPU 有十种寻址方式，可适应不同种数据访问方式，寻址方式决定了 CPU 执行指令的方法。

- 隐含寻址
- 立即寻址
- 直接寻址
- 扩展寻址
- 无偏移量变址
- 8 位偏移量变址
- 16 位偏移量变址
- 相对寻址
- 位操作寻址

- 位测试复合寻址

2.13.1.1 隐含寻址

隐含寻址指令中没有操作数，有关信息已含在操作码中。如中断返回指令(RTI)和停止指令(STOP)，一些隐含寻址指定是对 CPU 寄存器操作如设置进位标志(SEC)和累加器增加指令(INCA)，隐含寻址不需要存贮器地址，指令长度为一个字节。

2.13.1.2 立即寻址

立即寻址指令包含一个数值,用来对累加器和变址寄存器的值操作，立即寻址指令不需要地址，指令长度为二字节，第一字节为操作码，立即数为第二字节。

2.13.1.3 直接寻址

直接寻址指令可以访问存贮器中最低的 256 个字节，指令长度为二字节。第一字节为操作码，第二字节是操作地址的低 8 位。在直接寻址指令中,CPU 直接把$00 加到操作地址的高 8 位。

2.13.1.4 扩展寻址

扩展寻址指令是一个三字节指令，可访问任何存贮器单元。第一字节为操作码，第二和第三字节为操作地址的高位、低位。

2.13.1.5 无偏移量变址

无偏移量变址指令长度为一字节。变址寄存器包含操作地址的低位，可访问存贮器的前 256 字节 CPU 直接把$00 加入高位，指令地址分布为$0000～$00FF。无偏移量变址指令经常用来移动表格指针或保存被频繁使用的 RAM 数据及 I/O 地址。

2.13.1.6 8 位偏移量变址

8 位偏移量变址指令长两字节。可访问存贮器前 511 个字节的存贮单元。CPU 将变址寄存器中的无符号数加上操作码后续字节一个无符号数，它们之和作为操作地址，这种指令可访问$0000---$01FF。8 位偏移指令用于从一个有 N 个元素表中选择第 K 个元素。此表可从前 256 字节中任何一个地方开始，可扩展到$01FF，K 数值存放于变址寄存器。指令中放表格的首地址。

2.13.1.7 16 位偏移量变址

16 位偏移量变址指令长三字节，可访问所有存贮器地址， CPU 变址寄存器中的无符号整数加上操作码后面的二字节无符号数，它们之和作为操作地址。16 位偏移量变址指令也同样用于 N 个元素表中选择第 K 个元素。

2.13.1.8 相对寻址

只用于转移指令，发生转移时，操作码后的 8 位有符号整段加到 PC 上，否则顺序执行。偏移量为有符号整数，相对寻址范围为操作码地址开始的-128 到+127 个字节。

2.13.2 指令类型

此 MCU 共有 62 条基本指令，分为五大类

- 寄存器/存贮器指令
- 读－修改－写指令
- 跳 转指令
- 位操作指令
- 控制指令

2.13.2.1 寄存器/存贮器指令

大多数为双字节指令，其中一字节操作数在累加器或变址寄存器中，CPU 通过一种寻址方式从存贮器中找到另一操作数。表 9-1 列出了寄存器/存贮器指令。

表 9-1 寄存器/存贮器指令

指　　令	助　记　符
A 加存贮器和进位送 A	ADC
A 加存贮器送 A	ADD
A 和存贮器相“与”后送 A	AND
A 同存贮器位测试(逻辑“与”)	BIT
A 与存贮器算术比较	CMP
X 与存贮器算术比较	CPX
A 与存贮器异或　后送 A	EOR
从存贮器装载 A	LDA
从存贮器装载 X	LDX
乘　　X: A ← X*A	MUL
A 和存贮器相“或”后送 A	ORA
A 减存贮器及借位　后送 A	SBC
A 存入存贮器	STA
X 存入存贮器	STX
A 减存贮器　后送 A	SUB

2.13.2.2　读---修改---写指令

这类指令读取存贮器和寄存器中的值，修改它的内容，然后把修改的内容写回到存贮器和寄存器中。零或负测试 TST 指令，指定不属

于读－修改－写指令。因为它并不把修改的内容写到原来的存贮器中，表 9-2 列出了读---修改---写指令。

表 9-2 读---修改---写指令

指　　令	助　记　符
算术右移	ASR
清除	CLR
求反	COM
减少	DEC
增加	INC
逻辑左移	LSL
逻辑右移	LSR
(二进制)补码	NEG
带进位位循环左移	ROL
带进位位循环右移	ROR
负或零测试	TST

2.13.2.3 跳转指令（转移指令）

该类指令用于在某种条件具备时，使程序发生转移。该类指令均为相对转移指，均为双字节指令，除 BSR 为 6 时钟周期外，其他均为 3 个时钟周期。

指令	**功能**	**备注**
20 BRA Rel	(PC)+$0002+Rel → PC	总是发生转移
21 BRN Rel	(PC)+$0002→ PC	相当于 2 字节的 NOP

指令	功能	备注
22 BHI Rel	若(C)+(Z)=0，则(PC)+$0002+Rel → PC 若(C)+(Z)≠0，则继续	用于比较二进制无符号数的大小，大于则转移。
23 BLS Rel	若(C)+(Z)=1，则(PC)+$0002+Rel → PC 若(C)+(Z)≠1，则继续	用于比较二进制无符号数的大小，小于等于则转移
24 BCC Rel	若(C)=0，则(PC)+$0002+Rel → PC，否则继续	测试 C 值，为零则转移。用于比较二进制无符号数，
25 BCS Rel	若(C)=1，则(PC)+$0002+Rel → PC，否则继续	测试 C 值，为 1 则转移。
26 BNE Rel	若(Z)=0，则(PC)+$0002+Rel → PC，否则继续	测试 Z 值，为零则转移。
27 BEQ Rel	若(Z)=1，则(PC)+$0002+Rel → PC，否则继续	测试 Z 值，为 1 则转移。
28 BHCC Rel	若(H)=0，则(PC)+$0002+Rel → PC，否则继续	测试 H 值，为零则转移。
29 BHCS Rel	若(H)=1，则(PC)+$0002+Rel → PC，否则继续	测试 H 值，为 1 则转移。
2A BPL Rel	若(N)=0，则(PC)+$0002+Rel → PC，否则继续	测试 N 值，为零则转移。
2B BMI Rel	若(N)=1，则(PC)+$0002+Rel → PC，否则继续	测试 N 值，为 1 则转移。
2C BMC Rel	若(I)=0，则(PC)+$0002+Rel → PC，	测试 I 值，为零则转移。

指令	功能	备注
	否则继续	
2D BMS Rel	若（ I ）=1， 则 (PC)+$0002+Rel → PC，否则继续	测试 I 值， 为 1 则转移。
2E BIL Rel	若 IRQ_=0， 则 (PC)+$0002+Rel → PC， 否则继续	测试外部中断引脚， 为零则转移。
2F BIH Rel	若 IRQ_=1， 则 (PC)+$0002+Rel → PC， 否则继续	测试外部中断引脚， 为 1 则转移。

2.13.2.4 位操作指令

CPU 可对存贮器前 256 字节任一可读/写位执行置位或清零。包括端口寄存器，数据方向寄存器，定时器寄存器，以及地址在前 256 字节的 RAM。此 256 个单元的任何一位的条件特征状态都可以由软件测试和分支。位操作指令用直接寻址方式。表 9-4 列出了这些指令。

表 9-4 位操作指令

指令	功能	字节	周期
BSET n,addr8	将（addr8）的第 n 位置 1	2	5
BCLR n,addr8	将（addr8）的第 n 位置 0	2	5
BRSET n,addr8，Rel	若 (addr8) 的第 n 位为 1，则 (pc)+$003+Rel → PC 否则继续	3	5
BRCL n,addr8，Rel	若 (addr8) 的第 n 位为 0，则 (pc)+$003+Rel → PC 否则继续	3	5

2.13.2.5 控制指令

这些指令是寄存器参考指令，在程序执行时，用于控制处理器操

作,请参考下面控制指令表:

表 9-5 控制指令

指　　令	助　记　符
清进位	CLC
清中断屏蔽位	CLI
空操作	NOP
复位堆栈指针	RSP
从中断返回	RTI
从子程序返回	RTS
设进位位为 1	SEC
设中断屏蔽位为 1	SEI
停止	STOP
软件中断	SWI
传送 A 到 X	TAX
传送 X 到 A	TXA
等待	WAIT

2.13.3 指令查找表

表 9-6〈字母顺序列出了所有指令，及每个指令对状态寄存器的影响

指 令	操　×	功　能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
ADC #opr								IMM	A9	ii	2



指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
ADC opr								DIR	B9	dd	3
ADC opr	带 进 位	A ← (A)+(M)+(C)	*	–	*	*	*	EXT	C9	hh 11	4
ADC opr, X	加							IX2	D9	ee	5
ADC opr,X								IX1	E9	ff	4
ADC, X								IX	F9	ff	3
ADD #opr								IMM	AB	ii	2
ADD opr								DIR	BB	dd	3
ADD opr	无 进 位	A ← (A)+(M)	*	–	*	*	*	EXT	CB	hh 11	4
ADD opr, X	加							IX2	DB	ee	5
ADD opr,X								IX1	EB	ff	4
ADD, X								IX	FB	ff	3
AND #opr								IMM	A4	ii	2
AND opr								DIR	B4	dd	3
AND opr	逻辑与	A ← (A)∧(M)	–	–	*	*	–	EXT	C4	hh 11	4
AND opr, X								IX2	D4	ee	5
AND opr,X								IX1	E4	ff	4
ADD, X								IX	F4	ff	3
LSL opr								DIR	38	dd	5
LSLA								INH	48		2
LSLX	逻 辑 左		–	–	*	*	*	INH	58		2

指 令	操 ×	功 能	状态 H	 I	 N	 Z	 C	寻址 方式	操作码	操作 数	周 期
LSL opr,X LSL,X	移	C b7 b0 0						IX1 IX	68 78	ff	6 5
ASR opr ASRA ASRX ASR opr,X ASR,X	算 术 右 移	b7 b0 C	–	–	*	*	*	DIR INH INH IX1 IX	37 47 57 67 77	dd ff	5 2 2 6 5
BCC rel	进 位 位 为 0- 移	PC ←(PC)+2+rel? C=0	–	–	–	–	–	REL	24	rr	3
BCLR n opr	位清零	Mn ←0	–	–	–	–	–	DIR(bo) DIR(b1) DIR(b2) DIR(b3) DIR(b4) DIR(b5) DIR(b6) DIR(b7)	11 13 15 17 19 1B 1D 1F	dd dd dd dd dd dd dd dd	5 5 5 5 5 5 5 5
BCS rel	进 位 为 1- 移	PC←(PC)+2+rel, C=1	–	–	–	–	–	REL	25	rr	3
BEQ rel	相 等 转	PC←(PC)+2+rel,	–	–	–	–	–	REL	27	rr	3

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
	移	Z=1									
BHCC rel	半进位为 0 转移	PC←(PC)+2+rel, Z=0	–	–	–	–	–	REL	28	rr	3
BHCS rel	半进位为 1 转移	PC←(PC)+2+rel, H=1	–	–	–	–	–	REL	29	rr	3
BHI rel	大于转移	PC←(PC)+2+rel, C∨Z=0	–	–	–	–	–	REL	22	rr	3
BCC rel	大于等于转移	PC←(PC)+2+rel, C=0	–	–	–	–	–	REL	24	rr	3
BIH rel	$\overline{IRQ}$=1 移	PC←(PC)+2+rel? $\overline{IRQ}$=1	–	–	–	–	–	REL	2F	rr	3
BIL rel	$\overline{IRQ}$=0 移	PC←(PC)+2+rel? $\overline{IRQ}$=0	–	–	–	–	–	REL	2E	rr	3
BIT #opr BIT opr BIT opr BIT opr,X BIT opr,X BIT,X	位测试累加器和存贮器内容	(A)∧(M)	–	–	*	*	–	IMM DIR EXT IX2 IX1 IX	A5 B5 C5 D5 E5 F5	ii dd hh ll ee ff ff p	2 3 4 5 4 3
BCS rel	小于转	PC←(PC)+2+rel	–	–	–	–	–	REL	25	rr	3

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
	移	C=1,									
BLS rel	小于或等于转移	PC← (PC)+2+rel, C∨Z=1	–	–	–	–	–	REL	23	rr	3
BMC rel	I=0 移	PC← (PC)+2+rel, I=0	–	–	–	–	–	REL	2C	rr	3
BMI rel	N=1 转移	PC←(PC)+2+rel, N=1	–	–	–	–	–	REL	2B	rr	3
BMS rel	I=1 移	PC← (PC)+2+rel, I=1	–	–	–	–	–	REL	2D	rr	3
BNE rel	不等分支转移	PC← (PC)+2+rel, Z=0	–	–	–	–	–	REL	26	rr	3
BPL rel	大于等于转移	PC← (PC)+2+rel, N=0	–	–	–	–	–	REL	2A	rr	3
BRA rel	无条件转移	PC ← (PC)+2+rel	–	–	–	–	–	REL	20	rr	3
BRN rel	2 字节的 NOP	PC← (PC)+2	–	–	–	–	–		21	rr	3
								DIR(bo)	10	dd	5

指 令	操 ×	功 能	状态 H	I	N	Z	C	寻址 方式	操作码	操作数	周期
BSETn opr	设置 n 位	Mn ← 1	–	–	–	–	–	DIR (b1)	12	dd	5
								DIR (b2)	14	dd	5
								DIR (b3)	16	dd	5
								DIR (b4)	18	dd	5
								DIR (b5)	1A	dd	5
								DIR (b6)	1C	dd	5
								DIR (b7)	1E	dd	5
BSR rel	转到子程序指令	PC ← (PC); push (PCL) SP ← (SP)-1; Push(PCH) SP ← (SP)-1 PC ← (PC)+2+rel	–	–	–	–	–	REL	AD	rr	6
BRCLR n, opr, Rel	位测试"0"转移	n 位=0, 则 PC← (PC)+3+rel 否则继续	–	–	–	–	*	DIR (bo)	01	ddrr	5
								DIR (b1)	03	ddrr	5
								DIR (b2)	05	ddrr	5
								DIR (b3)	07	ddrr	5
								DIR (b4)	09	ddrr	5
								DIR (b5)	0B	ddrr	5
								DIR (b6)	0D	ddrr	5

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
								DIR(b7)	0F	ddrr	5
BRSETn, opr, Rel	位 测试"1"转移	n 位=1，则 PC ← (PC)+3+rel 否则继续	–	–	–	–	*	DIR(bo) DIR(b1) DIR(b2) DIR(b3) DIR(b4) DIR(b5) DIR(b6) DIR(b7)	00 02 04 06 08 0A 0C 0E	ddrr ddrr ddrr ddrr ddrr ddrr ddrr ddrr	5 5 5 5 5 5 5 5
CLC	清进位位	C ← 0	–	–	–	–	0	INH	98		2
CLI	清中断屏蔽	I ← 0	–	0	–	–	–	INH	9A		2
CLR opr CLRA CLRX CLR opr, X CLR, X	清寄存器	M ← $00 A ← $00 X ← $00 M ← $00 M ← $00	–		- 0	1	–	DIR INH INH IX1 IX	3F 4F 5F 6F 7F	dd ff 	5 2 2 6 5
CMP #opr CMP opr								IMM DIR	A1 B1	ii dd	2 3

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
CMP opr	累加器	(A) − (M)	−	−	*	*	*	EXT	C1	hh ll	4
CMP opr,X	内容同							IX2	D1	ee ff	5
CMP opr,X	存贮器							IX1	E1	ff	4
CMP,X	内容比							IX	F1		3
	较										
COM opr		M ← ($\overline{M}$)						DIR	33	dd	5
COMA	寄存器	A ← ($\overline{A}$)						INH	43		2
COMX	内容取	X ← ($\overline{X}$)	−	−	*	*	1	INH	53		2
COM opr,X	反	M ← ($\overline{M}$)						IX1	63	ff	6
COM,X		M ← ($\overline{M}$)						IX	73		5
CPX #opr	变址寄							IMM	A3	ii	2
CPX opr	存器内							DIR	B3	dd	3
CPX opr	容同存	(X) − (M)	−		*	*	1	EXT	C3	hh ll	4
CPX opr,X	贮器内							IX2	D3	ee ff	5
CPX opr,X	容比较							IX1	E3	ff	4
CPX,X								IX	F3		3
DEC opr		M ← (M)−1						DIR	3A	dd	5
DECA	寄存器	A ← (A)−1						INH	4A		2
DECX	内容减 1	X ← (X)−1	−		*	*	−	INH	5A		2
DEC opr,X		M ← (M)−1						IX1	6A	ff	6

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
DEC, X		M ← (M)−1						IX	7A		5
EOR #opr								IMM	A8	ii	2
EOR opr	累 加 器							DIR	B8	dd	3
EOR opr	内 容 同	A ← (A)⊕(M)	–		- *	*	–	EXT	C8	hh ll	4
EOR opr, X	存 贮 器							IX2	D8	ee ff	5
EOR opr, X	内容异							IX1	E8	ff	4
EOR, X	式							IX	F8		3
INC opr		M ← (M)+1						DIR	3C	dd	5
INCA	存 贮 器	A ← (A)+1						INH	4C		2
INCX	或 寄 存	X ← (X)+1	–		- *	*	–	INH	5C		2
INC opr. X	器 内 容	M ← (M)+1						IX1	6C	ff	6
INC, X	加 1	M ← (M)+1						IX	7C		5
JMP opr								DIR	BC	dd	2
JMP opr								EXT	CC	hh ll	3
JMP opr, X	无 条 件	PC ← 跳跃地址	–				- –	IX2	DC	ee ff	4
JMP opr, X	跳转							IX1	EC	ff	3
JMP , X								IX	FC		2
JSR opr		PC←						DIR	BD	dd	5
JSR opr		(PC)+n(n=1, 2, 3)						EXT	CD	hh ll	6
JSR opr, X	跳 转 子	Push (PCL);	–				-	IX2	DD	ee ff	6

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
JSR opr,X JSR ,X	程序	SP←(SP)-1 Push(PCH); SP ← (SP)-1 PC ← 条件地址					–	IX1 IX	ED FD	ff	5 5
LDA #opr LDA opr LDA opr LDA opr,X LDA opr,X LDA,X	存贮器内容装载到累加器	A ←(M)	–		- *	*	–	IMM DIR EXT IX2 IX1 IX	A6 B6 C6 D6 E6 F6	ii dd hh 11 ee ff ff	2 3 4 5 4 3
LDX #opr LDX opr LDX opr LDX opr,X LDX opr,X LDX,X	存贮器内容装载到变址寄存器	X ← (M)	–		- *	*	–	IMM DIR EXT IX2 IX1 IX	AE BE CE DE EE FE	ii dd hh 11 ee ff ff	2 3 4 5 4 3
LSL opr LSLA LSLX LSL opr,X	逻辑左移	C ← [b7 … b0] ← 0	–		- *	*	*	DIR INH INH IX1	38 48 58 68	dd ff	5 3 3 6

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
LSL ,X								IX	78		5
LSR opr LSRA LSRX LSR opr,X LSR ,X	逻 辑 右 移	0 → b7 … b0 → C	–		- *	*	*	DIR INH INH IX1 IX	34 44 54 64 74	dd ff 	5 2 2 6 5
MUL	无 符 号 乘法	X: A ← (X) X (A)	0	–	–	–	0	INH	42		1 0
NEG opr NEGA NEGX NEG opr,X NEG,X	存 贮 器 或 寄 存 器 内 容 取补	M ← (M)=$00-(M) A ← (A)=$00-(A) X ← (X)=$00-(X) M ← (M)=$00-(M) M ← (M)=$00-(M)	–		- *	*	*	DIR INH INH IX1 IX	30 40 50 60 70	ii ff 	5 2 2 6 5
NOP	空操作		–	–	–	–	–	INH	9D		2
ORA #opr ORA opr ORA opr ORA opr,X ORA opr,X	累 加 器 内 容 同 存 贮 器 内 容 相 “或”送	A ← (X) ∨ (A)	–		- *	*	–	IMM DIR EXT IX2 IX1	AA BA CA DA EA	ii dd hh 11 ee ff	2 3 4 5 4

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
ORA, X	入累加器							IX	FA	ff	3
ROL opr ROLA ROLX ROL opr, X ROL , X	带进位位的循环左移	C ← b7 … b0 ←	–		- *	*	*	DIR INH INH IX1 IX	39 49 59 69 79	dd ff 	5 2 2 6 5
ROR opr RORA RORX ROR opr, X ROR , X	带进位位的循环右移	→ b7 … b0 → C	–		- *	*	*	DIR INH INH IX1 IX	36 46 56 66 76	dd ff 	5 2 2 6 5
RSP	复位堆栈指针	SP ← $00FF	–	–	–	–	–	INH	9C		2
RTI	中断返回	SP ← (SP)+1; Pull(CCR) SP ← (SP)+1; Pull(A) SP ← (SP)+1; Pull(X)	*	*	*	*	*	INH	80		8

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
		SP ← (SP)+1; Pull(PCH) SP ← (SP)+1; Pull(PCL)									
RTS	子程序返回	SP ← (SP)+1; Pull(PCH) SP ← (SP)+1; Pull(PCL)						INH	81		5
SBC #opr SBC opr SBC opr SBC opr,X SBC opr,X SBC,X	累加器内容减去存贮器的内容再减去进位位	A ← (A)－(M)－(C)	–		- *	*	*	IMM DIR EXT IX2 IX1 IX	A2 B2 C2 D2 E2 F2	ii dd hh ll ee ff ff	2 3 4 5 4 3
SEC	设置进位位	C ← 1	–	–	–	–	1	INH	99		2
SEI	关中断	I ← 1	–	1	–	–	–	INH	9B		2
STA opr STA opr	将累加							DIR EXT	B7 C7	dd hh ll	3 4

指 令	操 ×	功 能	状态 H	I	N	Z	C	寻址方式	操作码	操作数	周期
STA opr,X STA opr,X STA,X	器内容保存到存贮器	M ← (A)	–	–	*	*	–	IX2 IX1 IX	D7 E7 F7	ee ff ff	5 4 3
Stop	使能 IRQ 中断,停止振荡器		–	*	–	–	–	INH	8E		2
STX opr STX opr STX opr,X STX opr,X STX,X	保存变址寄存器内容到存储器	M ← (X)	–	–	*	*	–	DIR EXT IX2 IX1 IX	BF CF DF EF FF	dd hh ll ee ff ff	3 4 5 4 3
SUC #opr SUB opr SUB opr SUB opr,X SUB opr,X SUB,X	累加器内容减存贮器内容送入累加器	A ← (A) – (M)	–	–	*	*	*	IMM DIR EXT IX2 IX1 IX	A0 B0 C0 D0 E0 F0	ii dd hh ll ee ff ff	2 3 4 5 4 3
		PC ← (PC)+1; Push(PCL)									

指 令	操 ×	功 能	状态 H	I	N	Z	C	寻址方式	操作码	操作数	周期
SWI	软件中断	SP ← (SP)-1; Push(PCH) SP ← (SP)-1; Push(X) SP ← (SP)-1; Push(CCR) SP ← (SP)-1; I ← 1 PCH ← 高位矢量中断 PCL ← 低位矢量中断	–	–	*	*	–	INH	83		10
TAX	传送累加内容到变址寄存器	X ←(A) (A)不变	–	–	–	–	–	INH	97		2
TST opr TSTA TSTX TST opr,X	测存贮器内容是负或0	(M) – $00	–	–	*	*	–	DIR INH INH IX1	3D 4D 5D 6D	dd ff	5 2 2 6

指 令	操 ×	功 能	状态					寻址方式	操作码	操作数	周期
			H	I	N	Z	C				
TST, X								IX	7D		5
TXA	传送变址寄存器内容到累加器	A ← (X)， (X)不变	-	-	-	-	-	INH	9F		2
WAIT	CPU 使能中断暂停 CPU	I=0	-	*	-	-	-	INH	8F		2

A.	累加器	opr	操作数(一或两字节)
C.	进位/借位标志	PC	程序计数器
CCR	状态寄存器	PCH	程序计数器高位
dd	直接操作地址	PCL	程序计数器高位
dd rr	直接操作地址和相对偏移转移命令	REL	相对寻址方式
DIR	直接地址方式	rel	相对程序计数器相对偏移字节
ee ff	址偏移量高、低位字节	rr	相对程序计数器相对偏移字节
	16 位偏移量变址	sp	堆栈指针

EXT	外部寻址方式	z	零标志
ff	址偏移量，8 位偏移量变址	#	立即数值
H	‰进位标志	∧	逻辑“与”
hh ll	外部寻址的高、低位操作地址	∨	逻辑“或”
L	中断屏蔽	⊕	逻辑异或
ii	立即操作数	( )	内容
IMM	立即寻址方式	-( )	负内容
INH	隐含寻址方式	←	装入到
IX	无偏移量变址寻址方式	?	如果
IX1	8 位偏移量变址寻址方式	:	连接
IX2	16 位偏移量变址寻址方式	↔	置位或清零
M	存储器地址	-	无影响
N	负标志	*	取决于指令执行结果
n	任何位	X	变址寄存器

3. 电路特性参数

3.1 最大额定值

PARAMETER	SYMBOL	VALUE	UNIT
电源	Vdd	-0.3—+7.0	V
输入电压	VIN	Vss-0.3 到 Vss+0.3	V
VPP	VIN		V
输出驱动电流（每 pin)	Id	25	mA
工作温度	TA	Tl-Th	°C

PARAMETER	SYMBOL	VALUE	UNIT
民用 工业		0 – 70°C – 40 – + 85°C	
存储温度范围	Tstg	-65 – +150°C	°C

3.2 直流电特性及定义:

Vss = 5.0Vdc±10%, Vss=0.0V, T=0-70°C

特性	符号	最小	典型	最大	单位
输出电压 I load= -10uA I load= +10uA	 Voh Vol	 Vdd-0.1 –	 – –	 – 0.1	 V V
输出高电压(Iload=-0.8mA) 全部 I/O 脚	Voh	Vdd-0.8	–	–	V
输出低电压(Iload=1.6mA) 全部 I/O 脚	Vol	–	0.1	0.4	V
输出高电平电流 Voh=2.5V 全部 I/O Voh=3V(PB5-PB7 低电流选择)	Ioh Ioh	10 2	– –	– –	mA
输出低电平电流 Voh=2.5V 全部 I/O Voh=3V(PB5-PB7 低电流选择)	Iol Iol	10 2			mA
总的 I/O 驱动电流	Iport	–	100	–	mA

特性	符号	最小	典型	最大	单位
输入高电平 PA0－1，PB0－1 IRQB，RESETB OSC1	Vih	0.7×Vdd	–	Vdd	V
输入低电平 PA0－1，PB0－1 IRQB，RESETB OSC1	Vil	Vss	–	0.2× Vdd	V
功耗 RUN(运行) Wait(等待) Stop(停机)	Idd	 – – –	 5.0 1.3	 9.0 2.5 3.0	 mA mA uA
I/O 高阻时漏电	Iil	–	–	±10	uA
输入电流 Timer ResetB,IrqB,OSC1	Iin	–	–	±1	uA
上拉电阻 PA0－1，PB0－7 PC0－7，PD0－7 ResetB,IrqB	 Rpu	 14 14 85	 36 36 100	 50 50 176	 KΩ

3.3 交流电特性及定义

VDD＝5.0Vdc±10%，VSS=0.0V， T=0-70°C

特性	符号	最小	最大	单位

特性	符号	最小	最大	单位
工作频率 RC 振荡选择 晶体振荡选择 外部时钟选择	Fosc	 0.1 0.1 直流	 4.0 4.0 4.0	 Mhz
内部工作频率 RC 振荡 晶体振荡 外部时钟	Fop	 – – –	 2.0 2.0 2.0	 Mhz
内部周期时间(1/Fop)	Tcyc	500	–	ns
RC 振荡器稳定时间	Trcon	–	1	ms
晶体振荡器启振时间	Toxov	–	100	ms
Stop 唤醒时间(晶体振荡器)	同 Toxov	–	100	ms
ResetB 脉冲最小宽度	Tr1	1.5	–	Tcyc
Timer 分辨率	Tres1	4	–	Tcyc
中断脉冲 0 宽度(边缘触发)	Tilih	125	–	ns
中断脉冲周期	Tilil	19	–	Tcyc
PA0－7 中断脉冲 1 宽度(边缘触发)	Tilil	125	–	Tcyc
PA0－7 中断脉冲周期	Tilih	19	–	Tcyc
OSC1 脉冲宽度	T	90	–	ns
RC 振荡器频率稳定性				

特性	符号	最小	最大	单位
Fosc=2Mhz, Vdd=5v±10%	ΔFosc	–		
Ta=–-40°C–+85°C		–	±25	%
Ta=0°C–+40°C		–	±15	%

备注:

(1) Idd(Run) 是指 Fosc = 2.0MHZ，全部输入 PIN 为 0.2Vdc,无直流负载时电流。

(2)Stop Idd ：全部 I/O 口为输入， Vil=0.2Vdc,Vih=Vdd-0.2Vdc, Osc1=Vss.

3.4 时序图

4.BENCH TEST 项目

4.1 目的

本项目规定了 BL0938 电路的 BENCH TEST 和 QC 测试的技术指标。

4.2 适用范围

适用 BL0938 电路 BENCH TEST 和 QC 测试。

4.3 测试条件

BL0938 有三种工作方式：用户方式，自检方式和开发方式。下表显示三种方式时的工作条件。

RESETB	VPP	PB1	工作方式
↑5V	Vss 到 Vdd	Vss 到 Vdd	用户模式
↑5V	2×Vdd	Vdd	自检模式
↑5V	2×Vdd	Vss	开发模式

4.3.1 用户模式

BL0938 的正常工作方式是处于用户模式下，当 VPP 和 PB1 电压处于 Vss 和 Vdd 之间，RESETB 脚的触发信号上升沿将使器件进入该模式。

4.3.2 自检模式

BL0938 可通过最少的外围硬件和片上$1F00－$1FEF 掩模自检程序，为用户提供基本的器件功能检测。片内含 240 字节自检程序，目的是让用户可自己检验器件的功能是否正确。

当 VPP＝2×VDD，PB1＝VDD，RESETB 脚的触发信号上升沿将使器件进入自检模式。以后 PB1 仍可做其它用处。进入自检模式后，自检程序可对 I/O, Timer, RAM, ROM, A/D 和各种中断功能检测。

如果功能正常，PC0－PC3 上的 LED 会连续循环闪亮。如果某一功能失效，LED 将会停止在某一状态，显示某功能出错，列表如下:

PC3	PC2	PC1	PC0	功能失效
1	0	1	0	I/O
1	0	1	1	Timer
1	1	0	0	RAM
1	1	0	1	ROM
1	1	1	0	A/D
1	1	1	1	中断
0	1	1	1	－

以后循环

备注“0”灯亮，“1”灯灭。

用 HP16500A 逻辑分析仪测试自检程序执行时，LED 的 PATTE－RN

如下:

测试目标	时间	PC3	PC2	PC1	PC0	备注
PORTC	0.0078ms	0	1	0	1	
PORTC	0.0184ms	1	0	1	0	
PORTC	0.008ms	0	1	0	1	
PORTC	0.138ms	1	0	1	0	
PORTB		1	0	1	0	
PORTA		1	0	1	0	
PORTD		1	0	1	0	
Timer	14.6ms	1	0	1	1	
RAM	412.7ms	1	1	0	0	
ROM	42.2ms	1	1	0	1	
A/D	0.317ms	1	1	1	0	
INT 中断	0.0785ms	1	1	1	1	
-	0.003ms	1	1	1	0	以下循环

在测试机上的自检测试如果用连线来实现，其它功能测试就难以连续进行，如果用继电器来连接，LOAD 板比较麻烦，可靠性和稳定性难测，也费时，但也是一个可用的方案。连线的目地是为外部提供输入信号，同时运行自检程序，减少大量 PATTERN。

4.4.3 开发状态的测试:

当芯片的 VPP = 2×VDD，PB1 = VSS 时，按 RESETB 键，RESETB 触发信号由低变高时，芯片进入开发状态。PB0 – 7 和 PC0 – 7 变成 MCU 的地址总线输入输出和一些控制信号的输出。对应如下:

PIN No.	16	15	14	13	12	11	10	9
符号	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
输出内容	A12	A11	A10	A9	A8	Irq2re	F78-1	F79
输出内容	A12	A11	A10	INSRD	R/WB	Irq2re	F78-0	F79

备注: A8 - A12 —— 地址总线的高五位。

Irq2re ———— IRQ2 中断的请求信号。

F78 ———————— 内部总线时钟。

F79 ———————— 与 F78 相移 90°的信号。

INSRD ———— 内部取指令信号。

R/WB ———— 内部存储器读写信号。

PIN No.	32	31	30	29	28	27	26	25
符号	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
输出内容	A7	A6	A5	A4	A3	A2	A1	A0
I/O	D7	D6	D5	D4	D3	D2	D1	D0

备注 A7 - A0 ———— 地址总线的低八位(F78=1)。

D7-D0 —— 八位数据总线(F78=0), 当 R/WB=1 时，外部数据(指令)可以输入。当 R/WB=0 时，内部数据总线可以输出。

芯片进入开发状态后，其内部 RAM 和 ROM 关闭，可以用外部 RAM 代替，这样就能开发用户软件和编写测试软件，我们的工程测试设备就是应用该状态。

测试 CORE 和除自检以外功能，也是选用开发状态，但是要注意首先打开内部 RAM(将#$08 写入$07FE 地址)，应为测试机只能起外部

ROM 作用，提供指令和读出内部总线。测试 PATTERN 就是这样。

后附 PATTERN:

(1) 自检测试 PATTERN:

(2) 指令测试 PATTERN:

(3) Timer 外输入测试 PATTERN:

(4) ADC 内部 RC 振荡的测试 PATTERN:

(5) 客户程序识别 PATTERN:

(6) DC 参数的测试 PATTERN:

4.4 初测和成测规范

4.4.1DC 规范:

Vdd = 5.0Vdc±10%, Vss=0.0V, T=0-70 °C

特性	符号	最小	典型	最大	单位
输出电压 I load= -10uA I load= +10uA	 Voh Vol	 Vdd-0.08 -	 - -	 - 0.08	 V V
输出高电压(Iload=-0.8mA) 全部 I/O 脚	Voh	Vdd-0.7	-	-	V
输出低电压(Iload=1.6mA) 全部 I/O 脚	Vol	-	0.1	0.35	V
输出高电平电流 Voh=2.5V 全部 I/O Voh=3V(PB5-PB7 低电流选择)	 Ioh Ioh	 10.5 2.5	 - -	 - -	mA

特性	符号	最小	典型	最大	单位
输出低电平电流 Voh=2.5V 全部 I/O Voh=3V (PB5-PB7 低电流选择)	 Iol Iol	 10.5 2.5			mA
总的 I/O 驱动电流	Iport	–	100	–	mA
输入高电平 PA0－1，PB0－1 IRQB，RESETB OSC1	Vih	0.7×VDD	–	Vdd	V
输入低电平 PA0－1，PB0－1 IRQB，RESETB OSC1	Vil	Vss	–	0.2×Vdd	V
功耗 RUN (运行) Wait (等待) Stop (停机)	Idd	 – – –	 3.0 1.3	 4.5 2.5 300	 mA mA uA
I/O 高阻时漏电	Iil	–	–	±9.5	uA
输入电流 Timer ResetB, IrqB, OSC1	Iin	–	–	±0.9	uA
上拉电阻 PA0－1，PB0－7	Rpu	15 15	30 30	48 48	KΩ

特性	符号	最小	典型	最大	单位
PC0－7，PD0－7 ResetB, IrqB		86	110	110	

备注:

(1) Idd(Run)是指 Fosc=4.0MHz,全部输入 PIN 为 0.2Vdd,无直流负载时的电流。Vdd＝5.0V，Vpp＝12.0V。

(2) Stop Idd：全部 I/O 口为输入，Vil=0.2Vdd, Vih=Vdd-0.2Vdd,Osc1=Vss,Vdd=5.0V， Vpp=12.0V。

(3) 其它测试根据 DC 测试 PATTERN 进行。

4.4.2 功能测试项目

(1)自检测试 PATTERN(初测和成测)。

(2)指令测试 PATTERN(初测和成测)。

(3)Timer 外输入测试 PATTERN(成测)。

(4)ADC 内部 RC 振荡测试 PATTERN(成测)。

(5)客户程序识别的测试 PATTERN(成测)。

4.4.3 C 规范

Vdd=5.0Vdc， Vss=0.0Vdc,T=25°C.

特性	符号	最小	典型	最大	单位
输出电压 I load= -10uA I load= +10uA	 Voh Vol	 Vdd-0.1 –	 – –	 – 0.1	 V V
输出高电压(Iload= -0.8mA) 全部 I/O 脚	Voh	Vdd-0.8	–	–	V

特性	符号	最小	典型	最大	单位
输出低电压(Iload=1.6mA) 全部 I/O 脚	Vol	-	0.1	0.4	V
输出高电平电流 Voh=2.5V 全部 I/O Voh=3V(PB5-PB7 低电流选择)	Ioh Ioh	10.0 2.0	- -	- -	mA
输出低电平电流 Voh=2.5V 全部 I/O Voh=3V(PB5-PB7 低电流选择)	Iol Iol	10.0 2.0			mA
总的 I/O 驱动电流	Iport	-	100	-	mA
输入高电平 PA0－1，PB0－7 IRQB，RESETB OSC1	Vih	0.7×Vdd	-	Vdd	V
输入低电平 PA0－1，PB0－7 IRQB，RESETB OSC1	Vil	Vss	-	0.2×Vdd	V
功耗 RUN(运行) Wait(等待)	Idd	 - -	 3.0 1.3	 5 2.5	 mA mA

特性	符号	最小	典型	最大	单位
Stop(停机)		-		400.0	uA
I/O 高阻时漏电	Iil	-	-	±10.0	uA
输入电流 Timer ResetB,IrqB,OSC1	Iin	-	-	±1.0	uA
上拉电阻 PA0－1，PB0－7 PC0－7，PD0－7 ResetB,IrqB	Rpu	14 14 85	30 30 110	50 50 176	KΩ

备注:

(1) Idd(Run)是指 Fosc=4.0MHz，全部输入 PIN 为 0.2Vdd，无直流负载时电流。

(2) Stop Idd：全部 I/O 口为输入，Vil=0.2Vdd,Vih=Vdd-0.2Vdd,scl=Vss,Vdd=5.0V,Vpp=12.0V。

(3) 其它测试根据 DC 测试 PATTERN 进行。

4.4.4 能测试项目

(1)自检测试 PATTERN(初测和成测)。

(2)指令测试 PATTERN(初测和成测)。

(3)Timer 外输入测试 PATTERN(成测)。

(4)ADC 内部 RC 振荡测试 PATTERN(成测)。

(5)客户程序识别的测试 PATTERN(成测)。

4.4.5ESD 性能指标:

±1000v DC 测试，功能测试符合规范。

4.4.6 LATCH－UP 性能指标:

根据公司规范 “GROUND INPUTS” Pulse Lvl=100ma

4.5 电气参数测试原理图

5. 典型应用图

6. 产品有关规范及技术资料列表

(1) 新品质量认证流程规范 260－1009。

(2) 非 EEPROM 类新品 R0 认证规范 260-1010。

(3) BL0938 产品测试规范。

(4) BL0938 产品使用说明书。

(5) 余永权<<MOTOROLA MC68HC05SR3 单片微型机原理及开发应用>>-广东高等教育出版社 1995